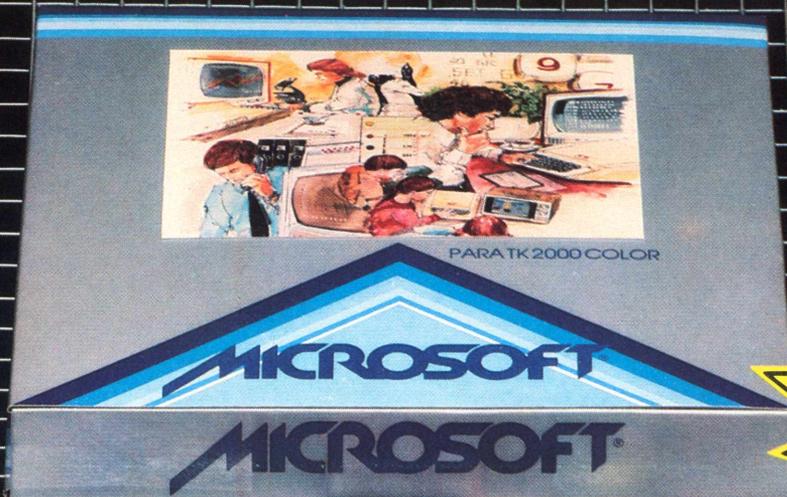
DICAS
TK 2000/APPLE

MICROHOBBY

Ano II - Nº 20 - 1985 Cr$ 6.000.

Para Manaus, Boa Vista, Porto Velho, Rio Branco, Santarém, Altamira, Macapá, Rondônia (Via Aérea): Cr$ 7.800.

USO DO COMPUTADOR:
Um problema de ética ou de conscientização?

Os melhores programas para você.
PARA TK 2000 COLOR
MICROSOFT
MICROSOFT
Garantia integral
MICROSOFT
MICROSOFT
MICROSOFT
MICROSOFT
A Microsoft tem 120 programas em fitas e disquetes à sua disposição.
São sistemas aplicativos para acompanhar e agilizar os negócios
de sua empresa. E também jogos eletrônicos para você e sua família
se divertirem muito. Todos especiais para TK-83, TK-85,
TK-2000, Apple II e compatíveis. E todos com a mesma qualidade
dos 100.000 programas já vendidos em todo o Brasil.
Procure o revendedor Microsoft mais próximo (se não encontrar os
programas Microsoft escreva para a Caixa Postal 54221 - CEP 01000-
S. Paulo-SP). Você encontrará os melhores programas da sua vida.
MICROSOFT
Sempre o melhor programa.

16 BITS É ATRAÇÃO NO MICRO-FESTIVAL

Solange Aparecida Menezes

Num espaço de 1.200 metros quadrados, 54 expositores deram uma amostra significativa de como está a área de informática no Brasil. O IV Encontro Brasileiro de Microinformática, o Micro-Festival/85, realizado no mês de março no Palácio das Convenções do Anhembi, foi organizado pela Guazzelli Associados, com patrocínio da Secretaria Especial de Informática — SEI e da SUCESU — Sociedade dos Usuários de Computadores e Equipamentos Subsidiários.

Houve vários lançamentos em microcomputadores comerciais e pessoais, periféricos, acessórios e software, mas a atração estava no lançamento dos micros 16 bits compatíveis com o PC da IBM. A Cobra apresentou seu micro 480; a SID, o micro 8005 multi-usuário, que suporta até cinco terminais e usa disco Winchester e sistema operacional Sinix; e mais quatro empresas apresentaram seus 16 bits compatíveis com o PC.

Um destaque foi a apresentação do MC-1000, terceiro microcomputador da CCE, que há três anos investe no mercado de micros. O MC-1000 possui CPU Z-80 a de 8 bits, duas saídas de vídeo incluídas na CPU, 22K de RAM expansíveis a 70K, efeitos sonoros e cor. Está previsto para breve a aceitação de até dois diskdrives.

A CCE e a Microdigital foram os estandes mais visitados pelo público mais jovem. Isso porque os videogames e jogos eletrônicos executados pelos micros continuam sendo as maiores atrações para os que não penetraram completamente no campo de informática.

Outro estande muito concorrido foi o da Compushop, freqüentado pelos mais entendidos em informática. Nele estava sendo apresentado o Harvard-PC, microcomputador de 16 bits compatível com o IBM-PC. Ele é considerado um micro revolucionário, pela empresa, pelo fato de permitir o back-up numa unidade de fita de meia altura, com capacidade de 10 Megabytes no mesmo gabinete. Mas não é só isso. O Harvard, fabricado pela Digicon, possui microprocessador 8088 totalmente compatível com o PC; 256K de memória RAM expansível até 1 Mbyte, monitor de vídeo alta-resolução padrão RGB, unidade de disco flexível slim 5.1/4 e disco rígido slim tipo Winchester.

No setor de software, o destaque foi o Super Calc^3, da Compucenter. Um programa integrado de planilha eletrônica, gráficos e gerenciamento de dados, indicado para resolver problemas financeiros, comerciais e matemáticos com rapidez e segurança. O Super Calc^3 é usado em equipamentos compatíveis com o IBM-PC, a planilha possui 127 colunas e 9999 linhas de capacidade teórica e 256K de memória. Além disso, o sistema é totalmente traduzido para o português.

O Micro-Festival/85 se estendeu por quatro dias e teve ainda 25 palestras, algumas delas destinadas à apresentação de softwares, como o Lotus 1, 2, 3, lançado no Brasil pela Sacco Computer.

84: ano de crescimento para Novadata

Operando no mercado de mini e superminicomputadores há aproximadamente três anos, a Novadata vem desenvolvendo um programa, desde novembro do ano passado, de aluguel de máquinas com o intuito de possibilitar acesso aos seus equipamentos, a um maior número de usuários. O programa prevê opção de compra pelos usuários em potencial e foi adotado devido ao alto preço das máquinas (os microcomputadores têm hoje um preço em torno de quatro milhões e setenta mil ORTNs), permitindo aos usuários, segundo representantes da empresa, avaliação e exploração dos minis sem maiores compromissos.

A empresa destacou-se em 83, conforme pesquisa publicada em 84 pela SEI — Secretaria Especial de Informática, pelo seu crescimento da ordem de 60%, considerado o maior do setor.

Conforme afirmou Mauro Farias Dutra, diretor superintendente, a Novadata fechou seu ano fiscal, referente a 84, com um lucro de 500 milhões de cruzeiros e atingiu faturamento bruto de oito bilhões e 500 milhões, com ingresso de recursos de um bilhão e 600 milhões captados entre os acionistas e na incorporação de lucros e correção monetária do capital. **A.L.A.**

Ciclo de Palestras anima o Micro-Festival/85

Solange Aparecida Menezes

Juntamente com as exposições houve um ciclo de palestras que mereceu grande atenção do público do Micro-Festival/85. Os temas apresentados foram: A microinformática no Brasil; Introdução à microinformática; Automação de escritórios; Oportunidades profissionais; A linguagem C, entre outros.

Mas o centro das atenções foi a apresentação do professor João Carlos Di Gênio, diretor-presidente das Faculdades Objetivo, que falou sobre o microcomputador no ensino. Muitas pessoas não participaram da palestra pela falta de espaço no auditório, mas o interesse dos presentes provou que a informática no ensino público e privado é uma preocupação geral.

As escolas Objetivo contam hoje com mais de 700 micros (CP-300, CP-500, Unitron, TRS-80 e Apple) sendo utilizados desde a pré-escola até a universidade. "No primeiro colegial o aluno aprende a linguagem LOGO como disciplina obrigatória, com uma hora de aula semanal", explicou Almir Brandão, também do Objetivo. Além disso, há também treinamento para os professores que, em sala de aula, manipulam o aparelho e os alunos acompanham suas explicações através de vários monitores espalhados pela sala.

O Objetivo espera contar, em breve, com os serviços de videotexto da Telesp e com um software integrado com uma rádio FM, onde seus alunos poderão estudar em casa, usando um microcomputador doméstico.

Outra novidade foi o seminário "O microcomputador, o profissional liberal e a pequena empresa", apresentado por Marcos Geribelo, diretor da Geribelo Engenharia. Não foi um seminário técnico dirigido a principiantes, mas o depoimento de um profissional liberal que empregou o computador para resolver seus problemas diários.

Engenheiro formado pela Politécnica de São Paulo, Marcos Geribelo nunca tinha tido contato com computadores antes de abrir, há três anos, sua empresa, que atua no ramo de construção civil e consultoria. "Senti necessidade de um instrumento de trabalho rápido quando trabalhava numa grande empresa", conta ele, "quando passei a trabalhar independentemente, surgiu a idéia de instalar um micro para oferecer um serviço melhor e mais rápido aos meus clientes".

Na Geribelo Engenharia todos os funcionários operam no microcomputador. "O computador não tem dono" disse ele, "todos os funcionários devem ter acesso ao equipamento".

Para os profissionais liberais presentes, Geribelo aconselhou a compra de um equipamento com a maior flexibilidade possível e a dispensa de qualquer uso de linguagem. "Quem entende do seu negócio é você e não um analista, portanto, você pode desenvolver seus próprios programas", disse.

No seminário sobre Processadores de texto na língua portuguesa, Eduardo Carvalho, gerente de suporte técnico da Compucenter e autor de um projeto para editores de texto na língua portuguesa, levantou a questão "se haverá espaço para a máquina de escrever no escritório automatizado". A conclusão foi positiva, mas apenas para remessa de pequenas correspondências.

Numa pesquisa realizada pela IBM, constatou-se que o uso de um processador de textos reduz o tempo e os gastos da empresa em 15%. Além desses benefícios, ele facilita a correção e alteração do texto, produz uma cópia de melhor qualidade, possibilita a experimentação, padroniza o trabalho e dá maior produtividade.

No encerramento do ciclo de palestras, Antonio Carlos Silveira, gerente da Abril Cultural, falou sobre as tendências de evolução dos microcomputadores e salientou o sucesso mundial do IBM-PC. Quanto ao contrabando de equipamentos, e a pirataria, Silveira foi taxativo: "esse problema só será resolvido quando o preço de mercado e a qualidade do programa forem equilibrados com a oferta clandestina".

Informática para leigos e iniciantes

Os principiantes e leigos em informática têm agora um núcleo para desenvolverem seus conhecimentos. A Sociedade dos Usuários de Computadores e Equipamentos Subsidiários — SUCESU, que completará 20 anos de atividades em 1986, criou o "Sucesu Micro-Clube" para oferecer oportunidade às pessoas que se interessam pela área.

Para isso foi criada uma infra-estrutura, com equipamentos de diversos fabricantes e instrutores para dar uma visão geral sobre informática ao novo usuário. Semanalmente serão realizadas reuniões, palestras e debates, de forma que os sócios possam trocar informações, esclarecer dúvidas e até decidir o tipo de equipamento mais adequado para suas necessidades.

Os usuários terão uma central permanente, onde poderão usar os equipamentos independente dos horários de reuniões. Além da prática, os sócios contarão com base teórica formada de publicações especializadas e documentos que serão preparados pela Sucesu.

Para se associar ao 'Sucesu Micro Clube' só é necessário ser pessoa física: profissionais liberais, usuários, estudantes, etc. — e não precisa possuir computador. A taxa de inscrição é de meia ORTN (baseado na ORTN de janeiro para inscrições feitas até junho e baseado na ORTN de julho para inscrições feitas no segundo semestre). "O volume de solicitações tem sido muito grande", informa Celso Furiani, diretor da Sucesu, que não soube precisar o número de novos sócios que se inscreveram no Micro-Festival/85, mas garante que "chegaremos a dois mil associados a curto prazo".

Aos interessados no "Sucesu-Micro-Clube", o endereço da Sucesu em São Paulo é rua Tabapuã, 627 — 1º andar; telefone 852-2144. E para os iniciantes, Celso Furiani recomenda: "entre no mundo da microinformática através do Micro-Clube. É o caminho mais rápido e econômico para você se capacitar nesse novo campo".

Programação de cursos e seminários

Maio

O Micro e suas Aplicações — 6 a 9/5 — Compushop — São Paulo — Infs.: (011) 815.0099/852.3360.

Redes de Teleprocessamento/ Conceitos e Aplicações — 15 a 17/5 — Servimec — São Paulo — Infs.: (011) 222.1511.

Robótica — Automação — Controle de Processos — CAD/CAM — 16/17/5 — Associação Brasileira de Administração Consumo Energia — São Paulo — Infs.: (011) 285.2490.

dBase Básico — 20 a 23/5 — Compushop — São Paulo. Infs.: (acima).

Introdução à Informática e Microcomputadores — 22/5 — Instituto de Tecnologia ORT — Rio de Janeiro — Infs.: (021) 286.7842.

Planilhas Eletrônicas — Idem acima.

Básico de Operação — 25/5 — ADP Systems — São Paulo — Infs.: (011) 223.4647.

Automação Comercial — 27/31/5 — Grupo Advancing — Porto Alegre — Infs.: (0512) 26.8242.

COBOL IBM — 27/5 — São Paulo — Idem acima.

Operação e Programação BASIC I — 28/5 — Mikro Informática — Minas Gerais — Infs.: (031) 222.3035/201.9754.

A Segurança Necessária em PD/ Criptografia — 27 e 28/5 — São Paulo — Servimec — Infs.: (Acima).

Micro-Mulher — 28/5 — São Paulo — Servimec — Infs.: (Acima).

Lotus 1-2-3 — 28 e 29/5 — São Paulo — Compushop — Infs.: (Acima).

O Método Warnier de Construção de Programas — 29 a 31/5 — São Paulo — Servimec — Infs.: (Acima).

Eletrônica Digital I — Metasoft — Rio de Janeiro — Infs.: (021) 220-1989.

BASIC I — São Paulo — SENAC — Infs.: (011) 255.0066.

Junho

Automação de Escritórios — 3 a 7/6 — Advancing — Porto Alegre — Infs.: (0512) 26.8246.

Análise de Sistemas — São Paulo — ADP Systems — Infs.: (011) 223.4647.

Exposoft/85.

Banco de Dados — 3 a 5/6 — São Paulo — Servimec — Infs.: (Acima).

Processador de Textos: Wordstar — 10/6 — Rio de Janeiro — Instituto de Tecnologia ORT — Infs.: (021) 286.7842.

Ligação Micro-Mainframe — 10 e 11/6 — São Paulo — Servimec — Inf.: (Acima).

Básico de Programação — 10/6 — São Paulo — ADP Systems — Infs.: (Acima).

Banco de Dados (dBase II) — 13/6 — Instituto de Tecnologia ORT — Rio de Janeiro — Infs.: (Acima).

Interfaces e Periféricos para Microcomputadores — 20 e 21/6 — São Paulo — Ass. Bras. Adm. Energia — Infs.: (Acima).

Centro de Informações e Transferência de tecnologia aos usuários — 19 a 21/6 — São Paulo — Compucenter — Infs.: (011) 255.5988.

Planejamento Estratégico e Metodologias em AE — 24 e 25/6 — São Paulo — 3I Informática — Infs.: (011) 521.9509.

Micromática apresenta novos produtos

Entre os novos lançamentos da Micromática Computadores e Sistemas está o módulo de Orçamento de Obra dirigido às empresas do setor de construção. Ao preço de 150 ORTNs, as empresas podem adquirir este módulo que apresenta relatórios dos serviços das obras, com ou sem BDI (margem de lucro), suas quantidades, preços unitários e valor total, fornecendo simultaneamente as mesmas informações a preços atuais ou projetados para recursos materiais, mão-de-obra e equipamentos que compõem esses serviços.

Este módulo é compatível com qualquer equipamento com CPM

Além dos serviços já citados estão incluídos nesse módulo: composições de serviços, tabela de preços, índices econômicos projetados para até dois anos e relação geral dos serviços e recursos. Além disso, a Micromática oferece outros módulos e sistemas que podem ser interligados entre sí.

Área de Recursos Humanos

Na área de recursos humanos, a Micromática lançou um sistema, o Memória, direcionado para o recrutamento e seleção de pessoal. Ele possibilita consultar, em tela ou em relatórios, informações sobre candidatos a emprego, cargos a serem preenchidos, normas administrativas para recrutamento, seleção e contratação de pessoal e sobre funcionários da empresa.

Ao preço de 250 ORTN, dividido por módulos, o Memória também emite cartas personalizadas e etiquetas de endereçamento.

O sistema operacional Memória é compatível com microcomputadores das linhas Apple, Itautec, Cobra, Unitron, IBMPC, de 8 e 16 bits. **S.A.M.**

Apple nacional com as qualidades do americano

O micro Appletronic 6502 fabricado pela Appletronic — Computadores e Sistemas — tem mantido as mesmas características do micro americano. A empresa, que antes de optar pela fabricação do Apple dedicava-se a manutenção dos Apple II, Apple II plus e III, optou pela manutenção da originalidade e qualidade do equipamento e utilização do mesmo material utilizado na fabricação do Apple original.

O Appletronic 6502 pode ser acoplado a diversos periféricos, entre estes, unidades de disco flexível 5 1/4" ou 8", máquinas de escrever eletrônica, plotters para o traçado de plantas e desenhos simples ou coloridos e modens para comunicação com outros micros.

Entre as aplicações específicas do micro, os responsáveis pelo seu desenvolvimento no Brasil destacam: simulação de um equipamento tipo Telex, utilização em controle de processos industriais através de interfaces tipo IEE 488, entre outras.

O micro está sendo comercializado atualmente ao preço de 120 ORTNs. **A.L.A.**

Novo horizonte para as pequenas e médias empresas

Organização e planejamento foram as palavras de ordem no seminário "A Informática e as pequenas e médias Empresas", realizado no dia 14 de março, no Hotel Transamérica, em São Paulo. Com a participação de quase 200 empresários, o seminário transcorreu durante todo o dia e teve a presença de figuras notáveis como Arnaldo Niskier, diretor da Bloch Educação; Guilherme Afif Domingos, presidente da Associação Comercial de São Paulo e Sérgio Araújo, presidente da Andei — Associação Nacional dos Dirigentes e Executivos de Informática, todos patrocinadores e organizadores.

O seminário, dividido em cinco apresentações, foi ministrado pelos executivos Lívio Antonio Giosa, diretor da Polidata; Rafael Ângelo Abud, diretor da Gaipe — Grupo de Apoio à Informática para a Pequena e Média Empresa; Marco Antonio Laurindo, diretor da Informarket Consultoria; Walter Lerner, diretor da Lerner & Associados e Ernesto Haberkorn, diretor da Siga — Sistemas e da Microsiga.

A Informática na pequena e média empresa

O objetivo deste seminário foi situar os empresários em relação ao que a informática pode oferecer às suas empresas e ajudá-los no processo de decisão. O que são hardware e software, o aparecimento da cibernética; o surgimento da necessidade do uso do computador para a expansão da economia da empresa foram alguns pontos básicos na primeira parte do seminário, onde empresários que estavam tomando contato com a área pela primeira vez puderam se familiarizar com a linguagem de computação.

Organização x implantação

"Se o empresário não se preocupar com a reestruturação da empresa, será inútil a aplicação do computador", afirmou Lívio Giosa. "Para a introdução do micro é necessário definir a cultura da empresa, isto é, qual o ambiente interno existente, para que o computador possa ser introduzido e venha a facilitar ou não o seu funcionamento".

Essa era uma idéia compartilhada por quase todos os conferencistas. Segundo eles, a organização e o planejamento da empresa são fatores primordiais para o sucesso da implantação do computador. "A motivação para a informática é o primeiro passo", explicou Rafael Abud. "Antes de pensar em informática deve-se ter uma estrutura. Antes de tudo, tem que ser observado o quadro de recursos humanos e fazer uma adaptação ao novo processo. Quando o indivíduo é valorizado no processo, ele é motivado para produzir mais", complementou.

Mas existem aqueles empresários tradicionais e resistentes, que não aprovam a introdução do computador em suas empresas. "A automação será aplicada pelo seu sucessor", exemplificou Abud, "geralmente seu filho, que atualmente está em contato com o computador na escola".

Os recursos e suas aplicações

Para os empresários que desejam implantar o microcomputador em suas empresas, Marco Laurindo ensinou os primeiros passos: "para chegar ao equipamento adequado deve-se fazer uma prévia do que se quer colocar no micro. Depois, procurar um bom profissional para saber qual o equipamento mais adequado; saber o significado de toda terminologia e procurar amigos possuidores de micros para trocar informações". Além destes fatores há também os softwares e periféricos.

Laurindo complementou afirmando que "o implante de um computador é resultado do planejamento e organização do setor administrativo da empresa. O custo da implantação é elevado, mas os resultados obtidos a médio prazo é bem grande". Essa idéia foi compartilhada por Walter Lerner. Para ele, mais de um milhão de empresas estão despreparadas para a implantação do computador. "O problema da empresa está vinculado à forma como ela está sendo administrada", disse ele.

Outro fator salientado pelos conferencistas foi quanto a automação dos departamentos. "Deve-se automatizar um de cada vez", explicou Laurindo, "e se começar sempre pelo mais fácil, para não tumultuar", concluiu. **S.A.M.**

Sempre o melhor programa para você

TELECOMUNICAÇÕES

- Programas para Projeto Cirandão
- Programas para Video-texto da Telesp
- Placas RS-232 da Arias Microcomunicações para TRS-80 e Apple
- Modens

SOFTWARE

O maior acervo de programas do Brasil que você pode: testar, usar, administrar, programar, desenhar e jogar livremente.

Disponíveis para as linhas: Apple, TRS-80 e Sinclair

HARDWARE

- CPU's das linhas: Apple, TRS-80 e Sinclair
- Interfaces para: Disco, Impressoras, CP/M, 80 colunas e Expansão de memória
- Drives para vários modelos
- Monitores e impressoras

SUPRIMENTOS

- Formulários contínuos
- Diskettes
- Etiquetas
- Fitas para impressoras

LIVROS

- Microproces. Z80 e 6502
- Cursos de Basic
- Programação estruturada
- Linguagens Basic, Cobol, Pascal
- Circuitos Eletrônicos
- Jogos Inteligentes
- Revistas

Av. Brigadeiro Faria Lima, 1390
8º And. Cj. 82 Tels.: (011) 813 6407 - 210 1251
01452 - J. Paulistano - São Paulo - SP

EDITORIAL

A nossa vida, a cada dia está cada vez mais sendo determinada pelos computadores. É o banco que mantém seu cadastro de cliente, é o microcomputador na sala de aula, é o videogame em sua sala de estar e é até a bomba de gasolina que verifica quantos litros e quantos cruzeiros você vai gastar.

Você já parou para pensar que um número grande de informações a seu respeito está nas mãos de profissionais que manipulam dados em computadores? Já se perguntou por que você tem recebido, pelo correio, malas-diretas de firmas que você nunca ouviu falar?

Dados "confidenciais" a seu respeito circulam como mercadorias entre empresas que processam dados, muitos deles fornecidos por você mesmo e outros conseguidos por outras vias.

Agora uma coisa para você parar e pensar: você mesmo tem acesso a estas informações a seu próprio respeito? Experimente conseguir saber porque seu cadastro não foi aprovado numa determinada instituição financeira.

Existe um outro profissional para quem damos informações confidenciais e confiamos nossas próprias vidas. É o médico. Isso nos parece muito natural, embora erros médicos sejam cometidos. Existe para nós uma certeza de que, pelo menos, há um código de ética entre eles, que determina o que podem ou não fazer.

Este código tem mais de dois mil anos e foi criado por Hipócrates, o pai de duas ciências: a Medicina e a Ética Profissional.

Não se trata aqui de fazer uma apologia da medicina, mas chamar a sua atenção para um problema que não pensamos, por comodismo ou até ignorância: existe uma Ética Profissional no uso do computador?

Como podemos garantir que os dados que fornecemos não serão vendidos ou até roubados por pessoas inescrupulosas?

E o que falar do software e da pirataria?

Como o profissional de informática poderá enfrentar estas condições sem algo que, vindo de sua própria classe, o oriente sobre o que ele pode ou não pode fazer?

A Sociedade como um todo necessita estar segura de que uma das maiores maravilhas que ela mesma criou se volte contra ela, não de uma forma Frankeinsteniana, mas pelos próprios homens que as manipulam.

Índice

Capa: Héctor Gómez Alísio

Expediente

DIRETOR RESPONSÁVEL
Szaya L. E. Seifert
ADMINISTRAÇÃO
Dijalma Peinado (gerente)
Marcia Regina Dominiquini
Marcia Augusto Esteves
EDITOR
Álvaro A. L. Domingues
JORNALISTA RESPONSÁVEL
Ana Lúcia de Alcântara Oshiro — Mt. 14.495
EDIÇÃO DE NOTÍCIAS
Ana Lúcia de Alcântara Oshiro
REDAÇÃO
Solange Aparecida Menezes (revisão)
Marcos Lorenzi
Fábio Polônio
ASSESSORIA TÉCNICA
Paulo Lauand
Wilson José Tucci
Aroldo Possuelo Carvalho
Angel D. Zaccaro Conesa
Gustavo Egídio de Almeida
DIAGRAMAÇÃO
Daniela S. Segre
CORRESPONDENTES
Rio de Janeiro — Fátima França
PUBLICIDADE
Aurio José Mosolino (supervisor)
Eduardo Garcia de Souza
ASSINATURAS
Siumara Farisco
CIRCULAÇÃO
José Aparecido Bueno
DISTRIBUIÇÃO
Fernando Chinaglia Distribuidora S/A
COMPOSIÇÃO E FOTOLITOS
Ponto Reproduções Gráficas Ltda.
IMPRESSÃO E ACABAMENTO
Editora Parma Ltda.

MICROHOBBY é editada mensalmente
por MICROMEGA PUBLICAÇÕES E
MATERIAL DIDÁTICO LTDA., INPI
2992 Livro A
Endereço para correspondência:
Caixa Postal 54096 — Fone: 255-0366
CEP 01296 — São Paulo, SP
Para solicitar assinaturas (12 números)
envie cheque nominal à MICROMEGA
P.M.D. Ltda., no valor de Cr$ 60.000

MICROHOBBY 20

CARTAS

Micro na Educação

Trabalho em Educação e o uso da informática vem ganhando mais e mais espaço, tanto que o próprio secretário da Educação já formou um comitê para aprofundar o assunto para a possível implantação do computador no setor educacional.

Como professor, adquiri recentemente um TK 85 e, de forma primária, tenho feito programas na área de Geografia (mapas, gráficos), da qual sou professor.

Gostaria de contar com o apoio de colegas que já elaboraram programas nesse setor e, se possível, gostaria de trocar programas e idéias como fim de aperfeiçoamento.

Como sugestão, seria interessante que outros professores fizessem o mesmo.

Avelino de Oliveira
Vargem Grande Paulista-SP

Caro Avelino,

É bastante oportuna sua carta, visto que nossa revista número 19 versou sobre o tema "O micro na Educação".

Nós, através da seção didática, temos publicado regularmente programas destinados à área educacional. Este espaço está aberto a colaborações e gostaríamos de contar com seus programas e de seus colegas. Seu endereço já foi publicado na seção Clube de Usuários da revista 19, mas, de forma a atender mais uma vez seu pedido, o republicamos aqui:
Avelino de Oliveira
Rua C, nº 11 — Jardim Marialva
06730 Vargem Grande Paulista-SP

Quanto aos outros professores, seria bom que os mesmos mandassem seus nomes e endereços para serem cadastrados no Clube de Usuários de forma a poderem ter um contato mais efetivo entre seus colegas de matérias e programação, independente da máquina utilizada.

TK 2000 x Apple

Em primeiro lugar, gostaria de parabenizá-los pela excelente revista que é a Microhobby, principalmente para usuários da família TK. O caminho está se abrindo cada vez mais para o TK 2000 e isso é bom, pois é difícil se encontrar literatura para ele.

Sou possuidor de um TK 2000 e observando o Apêndice E, "Palavras Reservadas" do Manual de Operações, descobri a função PDL(X). Procurei, então no manual e não encontrei nada a respeito. Peço, assim, uma explicação mais detalhada deste comando.

Além disso, lendo o artigo "Introduzindo o Ampersand", vi muitos endereços que executam determinadas funções (como FF69, FD1B, B1, DEC9, entre outros). Gostaria de saber quais os endereços destas funções no TK 2000.

Desde já agradeço a colaboração.

Alexandre Colaro
São Caetano-SP

Caro Alexandre,

Agradecemos seus elogios referentes à nossa atuação editorial, em relação ao TK 2000. Quanto às suas dúvidas, aqui estão as respostas:

a) A função PDL(X) lê a posição do paddle ou joystick analógico. Para poder usá-la é necessário uma interface especial que ainda não foi desenvolvida.

b) Os endereços que nos pede são:

Apple	*TK 2000*
FD1B	*FDOF*
FF69	*FF61*
B1	*B1*
DEC9	*CEEO ou D1E8*

A Volta do Barão Vermelho

Sou possuidor de um TK 2000 e fiquei feliz em ver uma revista com uma coluna destinada a este micro. Com respeito a esta coluna, venho pedir alguns esclarecimentos quanto ao programa "A Volta do Barão Vermelho", publicado na Microhobby número 16, no qual estou

com dificuldades em movimentá-lo através das setas, pois, no artigo consta que as linhas que o comandam são as de número 2040 e 2045, sendo que esta última não consta no programa. Também o som da explosão não se faz ouvir. Teria alguma instrução para este fim?

Robson Abrahão dos Santos
Niterói-RJ

Caro Robson,

Agradecemos sua preferência por nossa revista e a confiança que nela deposita.

Quanto a suas dúvidas, informamos que:

1) A linha 2045 é na realidade 2050.

2) A dificuldade em manobrar o avião é decorrente das seguintes características do jogo:

a) O avião do comandante Brow (em preto na tela) tem movimentos aleatórios.

b) O movimento deste avião se soma ao do Barão (cuja cabine aparece em vermelho na tela), que é pilotado pelo jogador. Como o avião do Barão serve como referencial, o jogador vê o avião inimigo se movimentando de acordo com a soma dos dois movimentos, como se o seu avião estivesse parado.

c) Como resultado, o movimento do joystick está aparentemente invertido. Se você movê-lo para a direita, o avião do Barão se move para a direita, mas como ele é seu referencial, você verá o avião inimigo se movimentar para a esquerda.

3) As rotinas de explosão poderão ser implementadas por meio da instrução Sound. Consulte seu manual para maiores detalhes.

TK DOS e DOS 3.3.

Eu sou possuidor de um TK 2000 e queria felicitá-los pelas novas seções para este micro, que encontrou seu primeiro espaço na Microhobby. Mas nunca é demais sugerir. Eu gostaria de que o TK 2000 tivesse as mesmas seções que os TK 83/85. Gostaria que vocês dessem ênfase ao BASIC e à linguagem de máquina do 6502.

Desejaria que os senhores respondessem às seguintes dúvidas:

a) Vocês pretendem criar livros para o TK 2000?

b) Para que servem as instruções OPEN e CLOSE, como operam no MBASIC?

c) Existem compiladores de linguagem como FORTRAN, PILOT, PASCAL, LOGO etc. para o TK 2000?

d) Quais são os comandos adicionais que fornece o TK DOS?

e) Quais as principais diferenças entre o DOS 3.3. e do TK DOS?

f) O TK DOS vem em disquete ou em um drive especial, criado por alguma empresa?

g) Como implementar o FLASH no TK 2000?

Jorge Pablo Zapata Rivera
Salvador-Bahia

Caro Jorge,

Agradecemos seus elogios e suas sugestões. Quanto aos livros sobre o TK 2000, informamos que no momento isso não está em nossos planos. Entretanto, todos os lançamentos da área merecerão divulgação em nossas páginas, na seção de livros.

As instruções OPEN e CLOSE abrem e fecham arquivos em disco. A instrução OPEN indica ao computador que foi aberto um arquivo em disco. Ela marca no disquete o início de um arquivo e seu nome. A instrução CLOSE fecha um arquivo, indicando seu final no disquete. Ela deve vir acompanhada do mesmo nome usado em LOAD.

No momento, ainda não existem compiladores para outras linguagens para o TK 2000.

O TK DOS fornece comandos de manipulação de arquivos. Entre eles estão o OPEN, o CLOSE, o CATALOG (fornece o nome de todos os arquivos), LOCK (tranca um arquivo, impedindo que seja apagado ou alterado), UNLOCK (destranca um arquivo); etc..

O TK DOS e o DOS 3.3 são muito semelhantes, permanecendo as diferenças do BASIC e do mapeamento de memória.

O TK DOS vem em disquete, mas precisa de um drive e de uma interface para ligá-lo ao TK 2000 I ou II. A interface é fornecida pela Microdigital e o drive pode ser qualquer um que funcione com APPLE.

O comando FLASH merecerá oportunamente um artigo para o TK 2000; aguarde.

CLUBE DOS USUÁRIOS

TK-85 e Compatíveis

Marcelo Ballario Yoshida
Av. Arthur C. Filho, 751
11660 — Caraguatatuba — SP
Área de interesse: contabilidade, filatélia, lazer.
Organiza um clube de usuários de TK-85 em sua cidade.

Jonas Eduardo Abboud
Av. Dr. Cavalcanti, 884
Fone: (011) 731-4316
13200 — Jundiaí — SP
Área de interesse: jogos, escuta de emissoras de rádio em ondas curtas.

Daniel Israel Mignone
R. Gomes Carneiro, 51
Ap. 104 — Ipanema
22071 — Rio de Janeiro — RJ
(também possuidor de TK 2000)

Aldo Jorge Nakamura
Av. Paulista, 854 — 12º andar
01310 — São Paulo — SP

Marcelo Alexandre de Souza
R. Barão da Torre, 32-B
AP. 907 — Ipanema
22411 — Rio de Janeiro — RJ
(possui alta-resolução)

Clayton Luiz Kohiyama
R. Maria Ortiz, 516
Bairro Campestre
09000 — Santo André — SP
(CP 200 SPEED)

TK-2000

Wladimir Barros de Assumpção
R. Dr. Vasconcelos, 162 — Centro
25800 — Três Rios — RJ
Área de interesse: jogos, aplicativos para engenharia eletrônica.

Carlos Henrique Pimentel
R. das Goiabeiras, 547
Bairro Jardim
09000 — Santo André — SP

APPLE

Eduardo Marchiori
R. Mariano Procópio, 58
Fone: 63-6258
01548 — São Paulo — SP
(APPLE II plus — possui programas e dicas sobre TK 85).

O SENTINELA

Bernardo C. Stein

Segurança é a palavra do momento. Nós, aficionados do TK, também temos que tê-la.

O programa "O SENTINELA" dá segurança total a seus programas em BASIC, não permitindo a ninguém acesso a eles, sem que se conheça o código secreto de proteção.

Suponhamos que você queira proteger o programa "BATE-CORAÇÃO", do colega Martello.

Para isso, você deve proceder da seguinte maneira:

1) "LOAD" o programa no computador.

2) Entre com uma linha 1 REM de 41 caracteres (figura 1).

3) Entre com as linhas do programa da figura 2, deletando as linhas 9998 e 9999 do programa original.

```
   1 REM 000000000000000000000
0000000000000000000
```
Fig. 1

```
9000 FOR N=16514 TO 16554
9010 PRINT N;"...";
9020 INPUT P
9030 PRINT P
9040 POKE N,P
9050 NEXT N
```
Fig. 2

4) Entre com GOTO 9000 (evite entrar com RUN 9000, pois isso iria apagar os possíveis dados do programa).

5) Entre com os códigos da figura 3.

```
16514...205
16515...145
16516...64
16517...254
16518...57
16519...32
16520...249
16521...205
16522...145
16523...64
16524...254
16525...48
16526...32
16527...242
16528...201
16529...205
16530...187
16531...2
16532...44
16533...40
16534...250
16535...45
16536...68
16537...77
16538...205
16539...189
16540...7
16541...58
16542...39
16543...64
16544...190
16545...40
16546...238
16547...126
16548...50
16549...39
16550...64
16551...201
16552...128
16553...128
16554...128
```
Fig. 3

6) Elimine as linhas 9000 e 9050 do programa, pois agora não serão mais necessárias (entre 9000 e NEW LINE, 9010 e NEW LINE e assim por diante).

7) Entre com as linhas indicadas na figura 4.

```
1510 PRINT AT 11,5;
"PROGRAMA PRO
TEGIDO"
1520 RAND USR 16514
1530 CLS
1540 GOTO 1
```
Fig. 4

8) Pronto. É só. Vejamos agora o resultado:
Entre com um "GOTO 1510".

9) Aparecerá a mensagem em letras invertidas "PROGRAMA PROTEGIDO" e nada conseguirá fazer o programa rodar.

10) QUASE NADA: experimente entrar com os caracteres "T K" em intervalos de mais ou menos meio segundo. Pronto! Proteção aberta (os caracteres de segurança são o "T" e o "K").

11) AGORA A PARTE MAIS IMPORTANTE: Você pode alterar os caracteres de segurança. Vejamos como:

12) Consulte o apêndice A do manual de seu TK: lá se encontra o conjunto de caracteres.

13) Suponhamos que você queira alterar o código para "B", "S":

14) O caractere "B" tem código 39.
O caractere "S" tem código 56.

15) Entre com os comandos diretos:
POKE 16518,39
POKE 16525,56

16) Experimente: entre "GOTO 1510" e tente rodar o programa. Você só conseguirá digitando "B" e depois o "S".

17) Agora grave seu programa dando "GOTO 1500". Depois disso você só irá rodá-lo digitando o seu código secreto.

Evidentemente você pode colocar "O SENTINELA" em qualquer programa em BASIC. Apenas tome os seguintes cuidados:

a) As linhas do programa que nós escolhemos de 9000 a 9050 devem ser colocadas nas linhas não usadas no seu programa principal.

b) As linhas da figura 4 devem ser colocadas após a instrução "SAVE . . ." do seu programa, para que ao se dar "LOAD" o mesmo vá automaticamente para a rotina em código de máquina (RAND USR 16514) e se bloqueie.

Anatomicro: A anatomia de um microcomputador

Compreenda o funcionamento, em linhas gerais, de um computador à base do Z 80 (o seu TK 85 ou TRS 80 modelo I ou II).

Tanios Hamzo

O PROCESSADOR

O gerente de todo microcomputador é o processador. Daí seu nome de Unidade Central de Processamento — CPU (Central Processor Unit).

O processador nada mais é do que um agrupamento de circuitos eletrônicos de lógica digital essencialmente simples. Acumulando funções de controle e de lógica aritmética e sendo capaz de acessar dados através de endereçamento próprio, ele pode realizar diferentes atividades no microcomputador a que pertence.

As operações que o processador é capaz de executar lhe são enviadas em forma de códigos binários digitais, muitas vezes combinadas umas com as outras para formar novas instruções. O conjunto de instruções de um processador não revela sua verdadeira versatilidade, pois depende da forma com que seu software pode ser desenvolvido e utilizado.

Pode ser um circuito puramente eletrônico, seu processamento é milhões de vezes mais rápido do que nossa concepção de rapidez: Seu "coração", dependendo do processador, pode bater a 1.000.000, 4.000.000 e até a 6.000.000 vezes por segundo. Isto significa, a grosso modo, que mesmo não sendo capaz de realizar grandes tarefas lógicas de cada vez, o processador pode realizar milhares de pequenas tarefas básicas e obter um complexo resultado de processamento em um curtíssimo espaço de tempo. Sua velocidade é a chave de sua utilidade e não a complexidade das tarefas que pode executar.

Um matemático, por mais medíocre que seja, pode conceber as mais simples operações (como a subtração, por exemplo); fazer contas desde dois algarismos até vários números, coisa que um processador comum não faz. Além disso, a velocidade dos fluxos elétricos de um processador é bem menor do que a do nosso cérebro (a máquina perfeita), mas ele pode executar as operações lógicas elementares mais facilmente e com lógica pura, tornando praticamente impossível a ocorrência de erros ou enganos de procedimento.

O Z-80

A CPU Z-80, mais especificamente o Z-80 A, é o processador que o TK 85 e o TRS-80 usa. Seu "clock" ou a freqüência com que seu "coração" bate, é de 4 MHz (4.000.000 batidas por segundo), variando levemente de modelo para modelo. No software, o Z-80, como qualquer outro processador, não compreende outra linguagem senão a de Máquina. Para fazer com que o Z-80 compreenda o BASIC é necessário que haja um programa capaz de "traduzir" as instruções do BASIC para o Assembly. É este o trabalho da ROM, que veremos em outra ocasião.

A aparência de um processador Z-80 é a de um grande CI com 40 pinos, 20 de cada lado, com funções específicas para controlar e ser controlado.

Todas as atividades de um Z-80 são interligadas ao mundo por, basicamente, quatro tipos de contato. São eles: alimentação elétrica; barramento de endereços; barramento de dados e controle.

Vejamos sua pinagem por partes semelhantes:

A) Alimentação elétrica:

A alimentação do Z-80 é simples, de uma única tensão de 5 volts bem regulados e estabilizados, mas tolerando pequenas oscilações de ruídos. Pela sua arquitetura física, o Z-80 tem uma boa imunidade a ruídos na alimentação e pode ser abastecido por fontes não muito sofisticadas. Os pinos 11 e 29 correspondem, respectivamente, à alimentação de 5 volts positivos e 0 volts (terra) compatível, inclusive, com fontes comuns para CI's de lógica TTL.

B) Barramento de endereços:

Com 16 bits, a barra de endereços é o maior barramento deste processador. Cada linha de endereço leva o código A, seguido do número correspondente (à potência de 2) que controla. Assim, as linhas do barramento de endereço são indexadas com números de zero a 15, conforme a tabela I.

Note que a nomenclatura corresponde mnemonicamente à pinagem e à potência de 2, o que torna mais fácil o manuseio e demonstra o requinte de fabricação. A letra "A" indica ser uma linha de endereço (Address) e com a combinação de níveis digitais entre to-

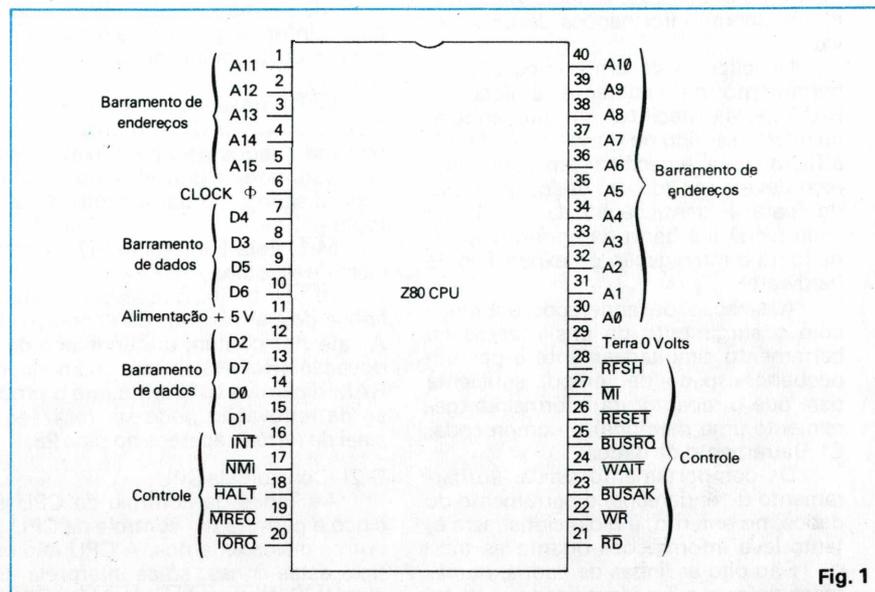



Fig. 1

Tabela I		
Linha de Endereço	**Grandeza**	**Pino**
A_0	$2^0 = 1$	30
A_1	$2^1 = 2$	31
A_2	$2^2 = 4$	32
A_3	$2^3 = 8$	33
A_4	$2^4 = 16$	34
A_5	$2^5 = 32$	35
A_6	$2^6 = 64$	36
A_7	$2^7 = 128$	37
A_8	$2^8 = 256$	38
A_9	$2^9 = 512$	39
A_{10}	$2^{10} = 1024$	40
A_{11}	$2^{11} = 2048$	1
A_{12}	$2^{12} = 4096$	2
A_{13}	$2^{13} = 8192$	3
A_{14}	$2^{14} = 16384$	4
A_{15}	$2^{15} = 32768$	5

das as 16 linhas, pode-se acessar qualquer endereço, desde zero (em binário = 0000000000000000) até 65535 (em binário = 1111111111111111). Isto significa que um Z-80 "só" pode endereçar 64 kBytes.

Estas 16 linhas de endereço trabalham em paralelo e, portanto, independentemente umas das outras. Quando reunidas são também chamadas de Address Bus (barramento de endereços em inglês).

A utilidade do endereçamento é que ele pode escolher (ou "acessar") outros dispositivos para serem processados, através do estado lógico de cada linha, selecionando, assim, um determinado código de localização a que chamamos de endereço (Address).

Determinando um endereço ao dispositivo, que geralmente é uma RAM ou algum tipo de ROM, a CPU passa a ter acesso ao conteúdo deste endereço, que é fornecido sob a forma de dados codificados em binário.

O fluxo de informações no barramento de endereço é unidirecional, isto é, só "saem" informações da CPU por ele.

No circuito de um computador, o barramento de endereços é ligado à RAM (para especificar em qual endereço deverá ser lido ou gravado um dado), à ROM (para especificar em qual endereço deverá ser lido um dado), ao teclado (para a identificação da função de cada tecla) e à barra de conexão externa (para a interligação de expansões de hardware).

A seleção de endereços acontece com o surgimento de uns e zeros no barramento simultaneamente e por um pequeno espaço de tempo, suficiente para que o receptor da informação (geralmente uma memória) o compreenda.

C) Barramento de dados:

De comportamento similar ao barramento de endereços, o barramento de dados, no entanto, é bidirecional, isto é, tanto leva informações quanto as traz.

São oito as linhas de dados, numeradas de zero a 7 e identificadas pela letra D (de Dado ou Data), da mesma forma que as linhas de endereço (veja a tabela I). **Tendo oito linhas, a Data Bus pode então estabelecer qualquer número entre zero e 255.**

O grupo de linhas de dados é também tratado de palavra ou byte. Assim, o Z-80 pode ser referido como um processador com byte de 8 bits.

A utilidade do barramento de dados é transportar dados paralelamente, entre dispositivos do computador, entre si, ou entre um dispositivo e a CPU. Este fluxo, como já disse, é bidirecional e num instante pode estar levando dados num sentido como no outro. Este fluxo pode ser entre dispositivos de entrada (como o teclado); dispositivos de saída (como a saída de vídeo); RAM; ROM; e a barra de conexão externa.

D) Pinos de controle

Os 14 pinos de controle são subdivididos em quatro sub-grupos: Controle de Sistema; Controle da CPU; Controle de barramento; e Clock.

Estes sub-grupos se prestam a funções de controle, tanto do processador por via externa, como do hardware externo pelo processador. Suas funções específicas são:

D.1) Controle do Sistema:

São seis os pinos para controle do sistema: $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{M1}}$ e $\overline{\text{RFSH}}$. Vamos a eles:

$\overline{\text{MREQ}}$: é a linha de Requisição de Memória (pino 19). O nível zero nesta linha significa que a CPU está requisitando determinado dispositivo (geralmente uma memória) para ler ou gravar os dados no barramento de dados, no endereço indicado pelo barramento de endereços.

$\overline{\text{IORQ}}$: linha de Requisição de I/0. Esta linha (pino 20), quando ativada, indica que a CPU está fornecendo um endereço pelos oito primeiros bits (pelo byte menos significativo) do barramento de dados, para leitura (entrada ou Input), ou para gravação (saída ou Output). Este sinal, em outras palavras, habilita uma porta de entrada/saída (I/0).

$\overline{\text{RD}}$: Leitura (Read). Pelo pino 21, a CPU informa que lerá dados de um endereço da memória ou de um dispositivo.

$\overline{\text{WR}}$: Gravação ou escrita (Write). Pelo pino 22, a CPU informa que está enviando dados pelo barramento de dados para serem gravados no endereço especificado pelo barramento de endereços.

$\overline{\text{M 1}}$: Esta linha (pino 27) informa o ciclo de Máquina 1.

$\overline{\text{RFSH}}$: Indica que as sete primeiras linhas do barramento de endereços (de A_0 até A_6) contém um endereço de renovação (ReFreSH) para memória tipo RAM dinâmica (DRAM) e que o processo de renovação pode ser realizado. O sinal de refresh aparece no pino 28.

D.2) Controle da CPU:

As linhas de controle da CPU são cinco e permitem o controle da CPU por outros meios externos. A CPU não controla estas linhas, só as interpreta. São elas: $\overline{\text{INT}}$, $\overline{\text{NMI}}$, $\overline{\text{HALT}}$, $\overline{\text{WAIT}}$ e $\overline{\text{RESET}}$.

Descrição:

$\overline{\text{INT}}$: Linha de Interrupção (pino 16). Interrompe o processamento da CPU, se esta não estiver executando nenhuma instrução e se a linha $\overline{\text{BUSRQ}}$ o permitir. Toda interrupção acatada pela CPU implica no surgimento de um sinal indicativo no início da instrução seguinte.

$\overline{\text{NMI}}$: Interrupção Não Mascarada (pino 17). De prioridade maior do que a linha $\overline{\text{INT}}$, esta linha força um reinício de processamento.

$\overline{\text{HALT}}$: É a única linha de controle da CPU que fornece uma saída, informando as ocasiões em que a CPU executou uma instrução $\overline{\text{HALT}}$ e aguarda por uma instrução de interrupção. Enquanto a CPU permanece "parada", vai executando automaticamente uma seqüência de instruções NOP, para que se mantenha a renovação das memórias. Esta saída acontece no pino 18.

$\overline{\text{WAIT}}$: Espera (WAIT). Ativada, informa à CPU que o dispositivo em uso (que pode ser uma memória, um dispositivo de entrada ou um dispositivo de saída) ainda não está pronto para uma transferência de dados. Desta forma, esta linha (pino 24) permite que um dispositivo mais lento do que a CPU (como uma impressora, por exemplo) esteja sempre em sincronismo com o processamento, acompanhando a CPU por mais lento que seja.

$\overline{\text{RESET}}$: O pino 26, quando ativado, provoca um RESET na CPU.

D.3) Controle de barramento:

São duas as linhas de controle de barramento:

$\overline{\text{BUSAK}}$: Quando em nível baixo, informa que o barramento de dados e o barramento de endereços da CPU estão ligados ao circuito. Quando em nível alto, informa que os barramentos estão "desligados" do circuito, na condição de alta impedância, não podendo assim receber ou transmitir dados. Pino 23.

$\overline{\text{BUSRQ}}$: Requisição de barra (BUS). Quando ativada, causa o estado de alta impedância nos barramentos de dados e de endereços da CPU, permitindo assim que outros dispositivos ligados possam usar os barramentos sem a interferência da CPU, que passa a se comportar como se não estivesse ligada ao circuito. As outras linhas de controle assumem o mesmo estado de alta impedância. Pino 25.

Os traços que encimam as siglas, significam que o pino é ativado quando em nível baixo.

D.4) Clock:

Ligado ao pino 6, a CPU necessita de um Clock. Pulsos de Clock são gerados por um circuito externo à CPU e temporizam os passos de processamento.

UM POUCO DE HISTÓRIA

Seguindo o rumo tecnológico do 8080, o Z-80, também um microprocessador de 8 bits, incorporou às vanta-

gens de seu antecessor mais 80 instruções de comando. Muitos dos circuitos externos que eram necessários aos processadores antigos foram incluídos dentro do mesmo encapsulamento do processador, tornando-o muito mais versátil e de simples montagem.

A tecnologia usada para a confecção deste processador é a de canal N e seu suporte não é exigente, com alimentação simples e clock de fase única. Todas as entradas e saídas são compatíveis com a lógica TTL.

Embora o Z-80 CPU tenha um amplo suporte da mesma família, (como o Z-80 CTC (Circuito Temporizador/Contador), o Z-80 PIO (I/0 Paralelo), o Z-80 DMA (Acesso Direto à Memória), o Z-80 SIO (I/0 Serial) e sistemas de apoio), não é usado nenhum deles no TK 85 e compatíveis.

O Z-80, embora compatível em muitos aspectos com 8080, não é compatível pino a pino. Hoje em dia o Z-80 já foi superado por, pelo menos, meia dúzia de outras famílias de microprocessadores, montados para tecnologias mais recentes e operando não com 8, mas com 16 e até 32 bits por byte. No entanto, a popularidade que o Z-80 ostenta ainda o deixará em cena por mais alguns anos (o que para um processador é muito tempo).

O SOFTWARE

O Z-80 é capaz de executar 150 instruções diferentes, inclusive 78 usadas pelo 8080, divididas em sete grupos diferentes: Carga e transferência; Transferência e procura de blocos; Aritmética e lógica; Manipulação de bits; Chamada e retorno; Salto e entrada/saída.

Ao todo, o Z-80 é capaz de reconhecer 696 códigos operacionais, dos quais 244 são também reconhecidos pelo 8080.

A ZILOG, primeira empresa a fabricar o Z-80, estabeleceu três velocidades de operação dos Z-80: 2 MHZ para o Z-80 propriamente dito; 4 MHZ para o Z-80 A; 6 MHZ para o Z-80 B.

Um exemplo:

As figuras de 2 a 6 mostram como podemos interligar as conexões de um Z-80 na construção de um micro computador.

Fig. 2

Fig. 3

Fig. 4

Todos os pinos de controle do sistema são acessíveis pelo conector de expansão.

Fig. 5

3 Só para modelos com duas 2732 EPROM. Para modelos com ROM de Kbytes ou para EPROM 2764, esta seleção é desnecessária.

** Para versões até 16 Kbytes de RAM. Para versões com mais de 16 Kbytes de RAM (32, 48 ou 56 Kbytes de RAM):
A0 A15

Fig. 6

HOBBYSHOP VEJA SE SUA CIDADE TEM O QUE VOCÊ PRECISA

SÃO PAULO

MICRO service

Inclusão de 24 novas funções (Read, Data, etc.), Slow, High Speed, Alta Resolução, Porta de I/0, etc. para micro de tecnologia SINCLAIR ZX81.
Manutenção de microcomputadores SINCLAIR (TK 82, 83, 85, etc.) e TRS.
Wilson de Assis — Tel.: 203-7967

TKSOM-TKMORSE

2 Software de alta qualidade para Micros Sinclair com 16 K
TKSOM — coloca som no seu micro; contém 6 músicas; você pode programar suas músicas.
TKMORSE — lista sua mensagem em código morse; transmite sinais sonoros de mensagem pré-gravada; transmite sinais sonoros simultaneamente com a digitação.
Preço até 30-06-85 Cr$ 28.000

Envie cheque nominal para:
MARCIO ACCIOLY
Rua Dr. Saboia de Medeiros, 199-54 — Cep 04120 — São Paulo — SP
e receba os 2 Software pelo correio, sem mais nenhuma despesa.
PREÇOS ESPECIAIS PARA REVENDEDORES.

Transforme seu TK 85.
O mesmo efeito dos monitores de vídeo.
Fundo: preto
Letras: brancas
Com uma simples modificação no microcomputador.

TRANSVIDEO Fone: (011) 522-8100

Comércio de Computadores Ltda.
TK85 x TK2000?
Só na ENG você adquiri o seu TK2000 nas melhores condições e ainda dá o seu velho TK83, TK85 ou CP200 como parte de pagamento. TK2000 é na ENG. Showroom — Tel. 813-7570.
Av. dos Pajurás, 40 — CEP: 05670.

Conheça os emocionantes jogos da **CIBERTRON** como os excelentes SPACE EGGS e GRAND PRIX por apenas Cr$ 16.900 cada. Faça seu pedido anexando cheque nominal cruzado, a CIBERTRON ELETRÔNICA LTDA. — Caixa Postal 17.005 — CEP 02399 — SP.

PROGRAME-SE!

Faça como os funcionários da SABESP, BURI, KIBON e outros. Venha desvendar o computador da DATA RECORD INFORMÁTICA.
COBOL — BASIC — DIGITAÇÃO
Turmas especiais para crianças de 8 a 14 anos. (BOLSAS DE ESTUDO)
Av. Santo Amaro, 5.450 — Tel. 543-9937 — Brooklin — (em frente ao E.C. Banespa).

QUAL A INTERFACE QUE ESTÁ FALTANDO NO SEU MICRO?

É AQUELA QUE LHE DEVOLVERÁ O PRAZER DE FICAR EM FRENTE DO SEU MONITOR POR TEMPO ILIMITADO.

MICROTELA possibilita que você continue com seu TV, pois possue a mesma tela de poliester utilizada nos monitores de última geração, filtrando e eliminando os reflexos, ao mesmo tempo que aumenta a resolução da imagem.
Adicionalmente proporciona o mesmo efeito repousante dos monitores de fósforo colorido, utilizando acrílico nas tonalidades verde e ambar.

Informações com **MASTER STING LTDA.**
Caixa Postal 18708 — São Paulo — SP

SUPRIMENTOS P/ INFORMÁTICA

*FORMULÁRIOS
*DISKETES *FITAS IMPRESSORAS
*PAPEL XEROGRÁFICO
*SUPRIMENTOS P/ TELEX E ESCRITÓRIO
INFORMAX-PRODUTOS P/ INFORMÁTICA LTDA.
R. Domingos de Morais, 254-6º and. Cj. 602-A
Tel. (011) 570.7570

SÃO JOSÉ DOS CAMPOS

EKTRONIC — COMPONENTES E SISTEMAS LTDA.

"**SOFT-LOADER**" — Interface micro-cassete para TK 82-C, 83, 85 e Ringo. Indica nível certo para carregar programas sem problemas e falhas. (Veja Microhobby Nº 10, 12 ou 13). Já um GRANDE SUCESSO PROVADO por centenas de usuários do TK. PREÇO: Cr$ 49.000,00 (Março).
Mande seu pedido com cheque nominal ou vale postal para EKTRONIC COMPONENTES E SISTEMAS LTDA. Caixa Postal 7004. São José dos Campos. CEP: 12200. Tel.: (0123) 291148.

BAHIA

Sua empresa poderia estar aqui.
Anuncie no HOBBYSHOP e todos os Leitores da região conhecerão sua empresa.
Anúncio econônico e de retorno garantido.

RIO DE JANEIRO

PROSERV-Processamento Dados.Cursos e Rep.Ltda.
.MICROCOMPUTADORES (Novos e Usados)
.CURSOS (Cobol. Basic. CP/M. DBase II)
.SUPRIMENTOS (Formularios. Disquetes. Fitas. etc.)
.LIVROS E REVISTAS
.SOFTWARE (TRS80. Apple. TK85)
Lg.Nove de Abril 27 salas 626/628
Tel: (0243) 429800 - V.Redonda - RJ

MINAS GERAIS

MICRO E VIDEO

Curso de Basic com turmas mensais
Programas para toda linha de microcomputadores — Sinclair, TRS-80, Apple, TRS Color, Comodore CP/M — Aplicativos e Jogos (Solicite catálogo especificando seu equipamento).
Livros e revistas nacionais e estrangeiros. Venda de Micros, periféricos e suprimentos. Soft House.
VILLABELLA SHOPPING — LOJA 6
Avenida Japão, 229 — Cariru — CEP 35160 — Fone (031) 821-2888 — Ipatinga — MG.

UM ARTIGO SÉRIO

Atenção! Não deixe de ler este texto até o fim. Ele é um texto sério onde a SURPRESA e o - CIV serão explicados e/ou justificados, dependendo do seu bom humor.
Em todos os textos que eu produzi para a Microhobby, sempre confiei bastante no bom humor e no espírito esportivo dos leitores.
Tenho certeza de que não é necessário, mas assim mesmo . . .

D E S C U L P E M - M E

. . . pela brincadeira da última edição.

Renato da Silva Oliveira

Certamente, alguns leitores perceberam que os artigos "π - α CIV, O Computador Inteligente" e "Surpresa" não passaram de brincadeiras.

Sem sombra de dúvida, estes leitores notaram que o π provém de Primeiro, o α provém de Abril, o CIV é 104 em algarismos romanos (1-04), e que as iniciais dos nomes dos cientistas imaginários formavam as palavras PRIMEIRO DE ABRIL.

Obviamente, estes leitores nem sequer deram-se ao trabalho de digitar o programa SURPRESA, pois logo perceberam que ele era, na verdade, um programa em BASIC.

Surpresa Visual

Vamos, agora, tentar obter algo de útil do programa SURPRESA: uma explicação de seu funcionamento aos que não o entenderam.

A memória RAM do TK é dividida em várias regiões. Os bytes entre 16384 e 16508, inclusives, são utilizados pela ROM para guardar as "variáveis do sistema". A partir do byte 16509, começa a região em que os programas são guardados (figura 1).

Figura 1

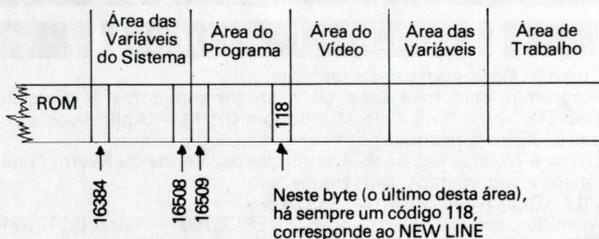
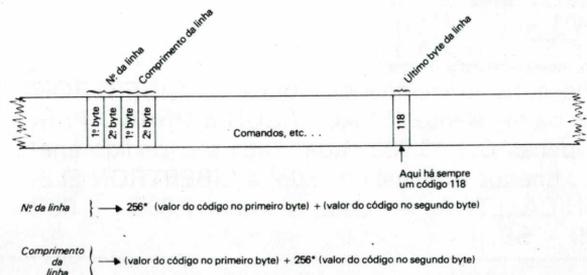
Esquema da RAM

Um programa é uma seqüência de linhas. Cada linha fica completamente determinada pelo seu número e pelo seu comprimento. Veja no esquema da figura 2 como essas informações são armazenadas (figura 2).

Figura 2

Armazenagem de uma linha de programa

OBS.: O comprimento da linha corresponde ao número de bytes usados pelos comandos, etc., mais o byte com o código 118.

Vamos analisar como o programa da figura 3 seria armazenado linha por linha conforme as figuras quatro à seis.

Figura 3

```
1 REM TESTE
2 PRINT "TESTE"
3 RUN
```

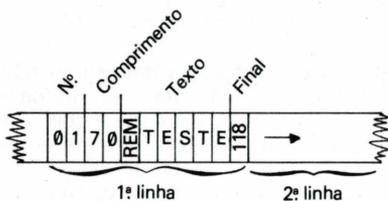
Figura 4

Número = Ø * 256 + 1 = 1
Comprimento = 7 + Ø * 256 = 7

Figura 5

Número = Ø . 256 + 2 = 2
Comprimento = 9 + Ø . 256 = 9

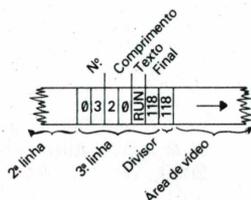
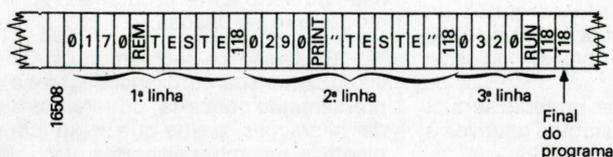



Figura 6

Número = Ø . 256 + 3 = 3
Comprimento = 2 + Ø . 256 = 2

Na memória, teríamos essas três linhas em seqüência (figura 7).

Figura 7

Agora, o que aconteceria se alterássemos o valor do comprimento da primeira linha para 26 (comandando POKE 16511,26)?

A resposta é fácil, as três linhas passariam a ser apenas uma: a linha 1.

Esse tipo de "truque" foi usado na confecção do programa SURPRESA, juntamente com outro "truque" que explicaremos adiante.

O micro distingue a área do vídeo da área do programa quando encontra dois códigos 118, um em seguida ao outro. O primeiro corresponde ao final de uma linha e o segundo ao final do programa. O micro sabe, então, que não deve listar mais nada a partir desses bytes.

Se nós quiséssemos fazer com que a listagem não aparecesse a partir de um certo ponto, bastaria introduzirmos dois códigos 118 em seqüência dentro de sua linha com REM.

Se comandássemos

```
10 POKE 16515,118
20 POKE 16516,118
```

a listagem no vídeo ficaria assim:

```
1 REM
```

A parte principal do programa SURPRESA estava em BASIC, comprimida numa só linha, a primeira.

Depois dessa linha, haviam duas outras com os comandos POKE necessários para abrir a primeira linha, isto é, separá-la em suas diversas linhas.

Haviam também comandos POKE para tirar os códigos 118 em seqüência, de modo a permitir a listagem do programa.

O último comando era o RUN, que fazia o programa "correr", a esta altura, já aberto.

O comando, RAND USR era só para despistar e, ao ser executado, o micro saia do BASIC e retornava imediatamente a ele.

Como fazer uma surpresa

Para juntar as várias linhas concentradas na linha 1, ao invés de contar o comprimento de cada linha (acrescido dos 4 bytes iniciais) e somá-los, foi usado um programa que realiza esse trabalho.

O programa a seguir é um "unificador" de linhas. Ao ser executado, ele aguarda a introdução de dois números de linhas (que devem necessariamente existir no programa). Introduzidos esses dois números, o programa UNIFICADOR junta todas as linhas entre essas duas (inclusive) numa só: a linha cujo número foi o primeiro a ser introduzido. Você não verá esta união na tela, mas experimente correr o cursor entre estas linhas, editar a primeira ou apagá-la.

Programa UNIFICADOR

```
600 REM UNIFICADOR
610 LET IP=0
615 PRINT "ENTRE A 1ª LINHA:"
```

```
620 INPUT LI
625 PRINT LI
630 PRINT "ENTRE A 2ª LINHA:"
640 INPUT LF
650 PRINT LF
655 FOR F=1 TO 30
656 NEXT F
657 FAST
660 LET I=16509
680 IF IP=1 THEN GOTO 730
690 IF PEEK I<>INT (LI/256) THE
N GOTO 860
700 IF PEEK (1+I)<>LI-256*INT (
LI/256) THEN GOTO 860
720 LET LI=I
725 LET IP=1
730 IF PEEK I<>INT (LF/256) THE
N GOTO 860
760 IF PEEK (I+1)<>LF-256*INT (
LF/256) THEN GOTO 860
780 LET I=I+PEEK (I+2)+256*PEEK
 (I+3)-LI
800 POKE (LI+3),INT (I/256)
820 POKE (LI+2),I-256*INT (I/25
6)
840 LIST PEEK (LI+1)+256*PEEK L
I
850 STOP
860 LET I=I+PEEK (I+2)+256*PEEK
 (I+3)+4
880 GOTO 680
```

A utilidade de um programa desse tipo não se resume apenas à elaboração de blefes como o programa SURPRESA.

Imagine uma situação em que você quer transformar as 30 primeiras linhas de um longo programa em suas 30 últimas linhas. Basta juntá-las e depois alterar o número da única linha resultante da junção. Por fim, deve-se reordenar os números das linhas, manualmente, ou com o auxílio de um programa ordenador.

Por aqui você já pode parar a leitura, pois o que havia para ser dito já o foi e . . . "Tudo que não se pode falar, deve-se calar". (L. Wittgeinstein)

Como Colaborar com Microhobby

Temos recebido várias cartas de pessoas interessadas em colaborar conosco, perguntando quais os critérios para publicação. Embora tenhamos usado, várias vezes, um anúncio da casa onde indicávamos como colaborar com Microhobby, resolvemos colocar neste número novas regras, de maneira a nos organizarmos melhor, garantindo tanto os interesses dos leitores, como os da revista Microhobby.

Assim, colocaremos algumas regras, que entrarão em vigor a partir desta publicação. As colaborações recebidas anteriormente serão analisadas, uma a uma, e entraremos em contato com os autores oportunamente.

Assim, ficam estabelecidas as seguintes regras:

a) Os autores que enviarem colaborações para Microhobby aceitam estas regras em carta anexa ao artigo.

b) Uma vez recebida a colaboração, fica implícito que seu autor autoriza a publicação de seu artigo.

c) Os artigos passarão por uma triagem e os autores dos artigos aceitos receberão uma comunicação onde constará: valor da remuneração estabelecido pela redação, segundo seus critérios; um contrato de cessão de direitos autorais que deverá ser assinado e devolvido à redação dentro de, no máximo, 10 dias. Qualquer discordância dos termos deverá ser comunicada neste período.

d) O pagamento será efetuado após a publicação do artigo, desde que o contrato tenha sido assinado. Caso o artigo já tiver sido publicado, o autor poderá assinar o contrato no momento do pagamento.

e) Os artigos não aceitos serão devolvidos aos autores (inclusive material anexo: fitas e listagens).

f) Os artigos e programas remunerados serão considerados propriedade da Micromega PMD Ltda., podendo esta fazer o uso que lhe convier (inclusive não publicá-lo).

g) As colaborações poderão ser:

1) Artigo sobre assuntos relativos à Informática de um modo geral;

2) Artigos sobre computadores de uma das seguintes linhas: TK 83/85, TK 2000, 2000 II, Apple ou seus compatíveis.

3) Programas para estas mesmas linhas. Junto com a listagem do programa deverá vir um texto explicativo sobre o funcionamento do mesmo, utilização, detalhamento, ou o que mais o autor achar conveniente. O computador utilizado deverá estar claramente indicado.

h) Programas acompanhados por fitas serão mais facilmente analisados.

i) O texto dos artigos ou programas deverão vir datilografados ou, pelo menos, escritos com letra legível.

j) Os autores asseguram a originalidade do texto e dos programas. As fontes deverão ser citadas.

Estes são os regulamentos. Todavia, quando seu autor manifestar-se expressamente por carta, poderemos estudar alterações, desde que sejam convenientes para ambas as partes.

SOMOS ESPECIALIZADOS EM INFORMÁTICA, ELETRÔNICA, MANUAIS DE FABRICANTES E DE CIRCUITOS, TELECOMUNICAÇÕES E ELETROTÉCNICA. PORTUGUÊS — INGLÊS — ESPANHOL.

LOTUS 1-2-3 — GUIA DO USUÁRIO

por Edward Baras

Guia prático para utilização e acompanhamento do software LOTUS 1-2-3, especialmente desenvolvido para as áreas financeiras e de negócios em geral.

Através de instruções simples, claras e de fácil compreensão, poderá atender suas necessidades de planejamento financeiro, aumentando sua capacidade de gerenciamento de informações e tomada de decisões. Cr$ 43.000

IBM PC E SEUS COMPATÍVEIS GUIA DO USUÁRIO

por Jonathan Sachs

Um guia prático para o usuário que deseja fazer seu IBM PC funcionar. O IBM-PC — Guia do Usuário, atende os usuários principiantes até os mais experientes, proporcionando um domínio completo do MICRO em pouco tempo.

O texto inclui o PC-XT e o sistema operacional DOS 2.0 que pode ser utilizado em todos os IBM-PC — compatíveis. Cr$ 39.000

MS-DOS — GUIA DO USUÁRIO

por Paul Hoffman/Tamara Nicoloff

Guia completo deste sistema operacional utilizado no IBM PC e em outros Micros de 16 bits.

Abrange todas as suas versões, inclusive a recentemente lançada 2-1, além de trazer informações completas sobre o PC-DOS — a versão exclusiva do MS-DOS para o IBM PC e seus compatíveis Cr$ 39.000

ZX-81 (TKs e compatíveis)

Hardware Sinclair — Para os micros TK 82C — NE

Hardware Sinclair — Para os micros TK 82C — NE Z8000 — TK83 — TK85 — CP200 — Lima .... Cr$ 18.000
Dissecando Jogos em Basic comentado linha por linha para TKs — Salvato .... Cr$ 20.900
Criando em Linguagem de Máquina — Ejchel .... Cr$ 33.200
Evoluindo no Basic TK — Stein .... Cr$ 29.800
O seu Micro e o Mundo Externo — Schon .... Cr$ 32.900

APPLE

20 Jogos Inteligentes em Applesoft — Para toda a Linha Apple — Palmer ..... Cr$ 15.000
Apple II Guia do Usuário — Poole/McNiff .... Cr$ 37.000
Manual de Basic — Para sistemas compatíveis com o Apple II — Peckham .... Cr$ 26.000
Programas usuais em Basic para Sistemas Compatíveis compatíveis como o Apple II — Poole .... Cr$ 22.000
Por Dentro do Apple — Tucci .... Cr$ 35.000
Dê um Apple e sua Vida — Mirshawka . Cr$ 36.000
Programas Comerciais da linha Apple para pequena empresa — Abreu ...... Cr$ 25.000
77 Programas para linha Apple — Abreu Cr$ 25.000
Conhecendo e utilizando o TK-2000 — Mirshawka .... Cr$ 18.000
TK-2000 na Matemática — Mirshawka . Cr$ 28.000
Usando o Visiplot — Abreu/Lima ..... Cr$ 21.000
A Primeira Mordida — Tucci/Moreira .. Cr$ 19.500
Elppa II Plus — Manual de Instruções — Victor do Brasil .... Cr$ 19.500

MICROCOMPUTADORES/ MICROPROCESSADORES

Microprocessadores e Microcomputadores — Hardware e Software — Tucci Cr$ 35.000
Introdução a Arquitetura e Organização de Computadores — Lorin .... Cr$ 36.000
Matemática para Microcomputadores — Barden Jr. .... Cr$ 18.000
Microcomputadores e Microprocessadores — Malvino .... Cr$ 27.900
Introdução aos Microcomputadores — Volume 0 — Osborne .... Cr$ 21.900
Microprocessadores — Conceitos Básicos — Volume 1 — Osborne .... Cr$ 24.900
Microprocessadores de 16 Bits — Titus/Titus .... Cr$ 47.900
Aplicação de Microprocessadores na Indústria — NCC .... Cr$ 46.800
Introdução aos Microcomputadores — Bianchi .... Cr$ 19.800
Introdução aos Microprocessadores — Tokheim .... Cr$ 24.900
Aplicações aos Microprocessadores — Kuecken .... Cr$ 36.800
Microcomputador e Informática — Shimizu .... Cr$ 15.000

PREÇOS SUJEITOS A ALTERAÇÃO

Atendemos pelo Reembolso Postal e VARIG, com despesas por conta do cliente, para pedidos acima de Cr$ 10.000,00 (VARIG: Cr$ 30.000,00). Pedidos menores devem vir acompanhados por cheque nominal ou Vale Postal, acrescidos de Cr$ 1.000,00 para as despesas de despacho pelo correio.

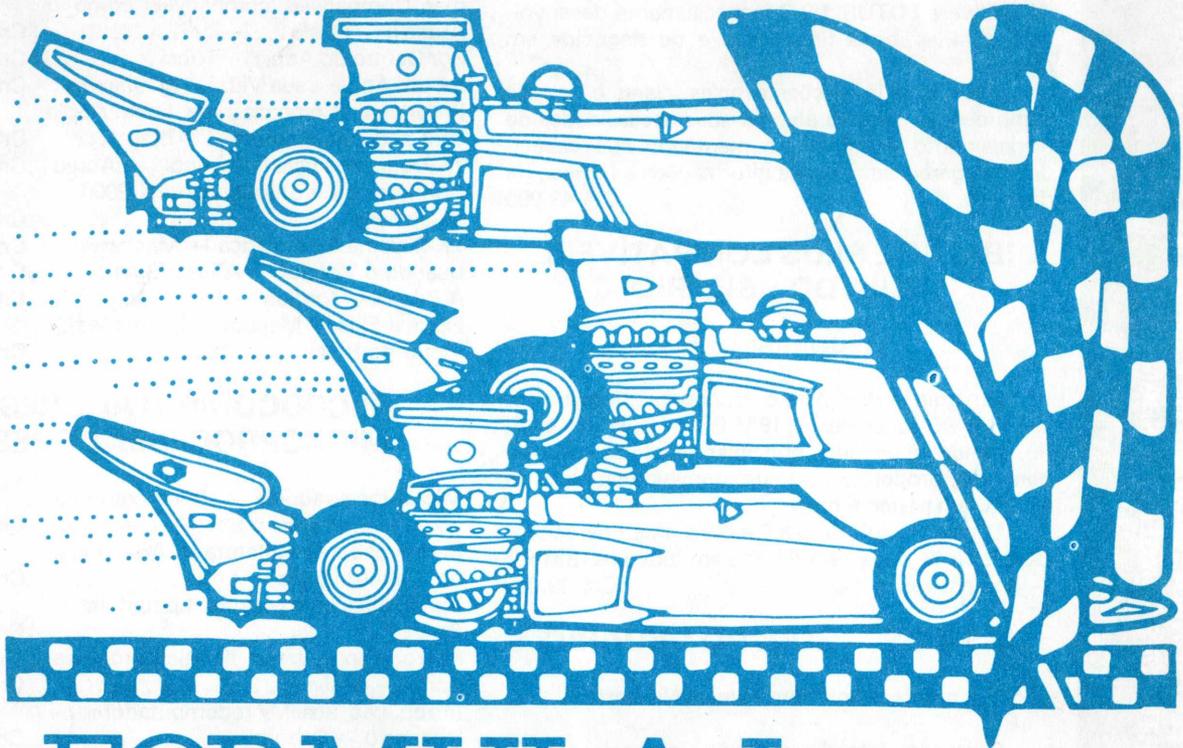
Litec

LIVRARIA EDITORA TÉCNICA LTDA.

RUA DOS TIMBIRAS, 257
01208 — São Paulo — SP
Tel.: 220-8983
Caixa Postal 30.869

Fórmula I
Memória Ocupada: 9463 bytes
Soma Sintática: 23141

Nível: (3 computadores)

Computadores: TK-83 e compatíveis

FORMULA I

Raul Udo Christmann

OBJETIVO

O objetivo deste jogo é levar um carro (representado por   ) do ponto de largada até o de chegada, num circuito desenhado na tela, no menor tempo possível e com o combustível disponível.

COMO ENTRAR COM OS DADOS

O circuito contém 52 posições e, à cada uma, deve-se informar o acréscimo de velocidade (mais ou menos) e a marcha (posição da alavanca da mudança, 1 a 5). Exemplos:

— 4Ø1 se for na saída, representa a partir da imobilidade para 40 km/h, utilizando a 1ª marcha.
— 3Ø2 diminui a velocidade de 3Ø km/h, em 2ª marcha.
Ø5 mantém a velocidade atual (acréscimo igual a zero), em 5ª marcha.

RESULTADOS PARCIAIS

Após cada indicação de velocidade, alguns resultados parciais são representados na tela:

a) velocidade do momento, em km/h;
b) velocidade média até o momento, em km/h;
c) tempo em segundos consumido até o momento;
d) rotação do motor em rpm (rotações por minuto);
e) marcha em uso no momento;
f) saldo de combustível, em litros;
g) condições da pista duas posições à frente;
h) avisos diversos.

→

```
   1 REM CORRIDA DE FORMULA-1
   2 REM POR R.UDO CHRISTMANN
  10 PRINT AT 10,5;"CORRIDA DE F
ORMULA-1";
  20 PAUSE 360
  30 CLS
  40 LET T$="DIGITE <NEW LINE>"
  50 CLS
  60 PRINT TAB 3;"""CORRIDA DE F
ORMULA -1"""
  70 PRINT
  80 PRINT TAB 3;"-A CADA INSTAN
TE, DIGITE"
  90 PRINT "  O ACRESCIMO DE VEL
OCIDADE:"
 100 PRINT "  (+OU-).E A MARCHA
(1 A 5)"
 110 PRINT "  PARA O PROXIMO TRE
CHO..."
 120 PRINT AT 20,6;T$
 130 INPUT G$
 140 CLS
 150 PRINT "  ""ATENCAO"", PARA
OS LIMITES:"
```

```
 160 PRINT AT 2,4;"-MARCHA V.MAX
.ACELER.-"
 170 PRINT ,,TAB 6;"1       90
+50/-50"
 180 PRINT TAB 6;"2       128    +4
0/-40"
 190 PRINT TAB 6;"3       270    +3
0/-30"
 200 PRINT TAB 6;"4       360    +2
0/-20"
 210 PRINT TAB 6;"5       450    +1
0/-10"
 220 PRINT
 230 PRINT TAB 5;"RPM MAX.DO MOT
OR=9000"
 240 PRINT TAB 5;"COMBUSTIVEL(LT
S)=500"
 250 PRINT AT 20,6;T$
 260 INPUT G$
 270 CLS
 280 PRINT "NAS CURVAS A V.MAX.F
ICA REDUZIDAEM 40 P/CENTO."
 290 PRINT
 300 PRINT "-VELOCIDADE  PROXIMO
DO LIMITE   CAUSA DERRAPAGEM E
ACRESCIMO   NO TEMPO (MULTA)."
 310 PRINT
 320 PRINT "-PISTA COM ""OLEO""
V.MAX.REDUZIDA EM 40 P/CENTO."
 330 PRINT
 340 PRINT "-PISTA""UMIDA"". V.M
AX.REDUZIDA     EM 10 P/CENTO."
 350 PRINT
 360 PRINT "-PISTA""MOLHADA"".V.
MAX.REDUZIDA    EM 20 P/CENTO."
 370 PRINT AT 20,6;T$
 380 INPUT G$
 390 CLS
 400 PRINT AT 20,6;"AGUARDE UM P
OUCO..."
 410 PAUSE 120
 420 CLS
 430 FAST
 440 DIM A(52)
 450 LET K=122
 460 FOR I=1 TO 11
 470 LET A(I)=K
 480 LET K=K-2
 490 NEXT I
 500 LET A(12)=302
 510 LET K=502
 520 FOR I=13 TO 18
 530 LET A(I)=K
 540 LET K=K+2
 550 NEXT I
 560 LET A(19)=712
 570 LET K=912
 580 FOR I=20 TO 25
 590 LET A(I)=K
 600 LET K=K-2
 610 NEXT I
 620 LET A(26)=1102
 630 LET K=1302
 640 FOR I=27 TO 33
 650 LET A(I)=K
 660 LET K=K+2
 670 NEXT I
 680 LET K=1215
 690 FOR I=34 TO 38
 700 LET A(I)=K
 710 LET K=K-200
 720 NEXT I
 730 LET A(39)=417
 740 LET K=419
 750 FOR I=40 TO 44
 760 LET A(I)=K
 770 LET K=K+200
 780 NEXT I
 790 LET A(45)=1320
 800 LET A(46)=1322
 810 LET K=1223
 820 FOR I=47 TO 52
 830 LET A(I)=K
 840 LET K=K-200
 850 NEXT I
 860 CLS
 870 LET A$="████████████████████
███"
 880 LET C$="█"
 890 PRINT AT 0,17;"SAIDA"
 900 PRINT AT 1,2;A$
 910 PRINT AT 2,3;A$( TO 20)
 920 PRINT AT 4,16;A$( TO 5)
 930 PRINT AT 5,3;A$( TO 10);TAB
 16;A$( TO 3)
 940 PRINT AT 6,2;A$( TO 10)
 950 PRINT AT 9,2;A$( TO 10)
 960 PRINT AT 10,3;A$( TO 10)
 970 PRINT AT 13,3;A$( TO 12);TA
B 20;A$( TO 3)
 980 PRINT AT 14,2;A$( TO 14);TA
B 19;A$( TO 5)
 990 FOR I=1 TO 5
1000 PRINT AT I,2;C$
1010 NEXT I
1020 FOR I=9 TO 13
1030 PRINT AT I,2;C$
1040 NEXT I
1050 FOR I=2 TO 4
1060 PRINT AT I,3;C$
1070 NEXT I
1080 FOR I=10 TO 12
1090 PRINT AT I,3;C$
1100 NEXT I
1110 FOR I=6 TO 8
1120 PRINT AT I,12;C$
1130 NEXT I
1140 FOR I=5 TO 9
1150 PRINT AT I,13;C$
1160 NEXT I
1170 FOR I=4 TO 12
1180 PRINT AT I,15;C$;TAB 20;C$
1190 NEXT I
1200 FOR I=5 TO 13
1210 PRINT AT I,16;C$;TAB 19;C$
1220 NEXT I
1230 FOR I=2 TO 12
1240 PRINT AT I,23;C$
1250 NEXT I
1260 FOR I=1 TO 13
1270 PRINT AT I,24;C$
1280 NEXT I
1290 PLOT 5,41
1300 PLOT 7,39
1310 PLOT 5,25
1320 PLOT 7,23
1330 PLOT 24,25
1340 PLOT 26,23
1350 PLOT 30,17
1360 PLOT 32,15
1370 PLOT 31,35
1380 PRINT AT 1,26;"C.GER."
1390 PLOT 33,33
1400 PLOT 46,17
1410 PLOT 48,15
1420 PRINT AT 2,26;"PISTA"
1430 PRINT AT 3,16;"CHEGADA";TAB
 26;"-----"
1440 PRINT AT 12,25;"PISTA:"
1450 PRINT AT 16,1;"VELOC:";TAB
12;"K/H ROTAC.";             TAB
27;"RPM"
1460 PRINT AT 17,1;"V.MED.";TAB
12;"K/H MARCH:";
1470 PRINT AT 18,1;"TEMPO:";TAB
12;"SEG.GASOL:";             TAB
27;"LTS."
1480 PRINT AT 20,0;"DIGITE O ACR
ESCIMO DE VELOCIDADEE A MARCHA P
/PROXIMO TRECHO."
1490 SLOW
1500 LET VMAX=450
1510 LET D=0
1520 LET POS=0
1530 LET P=0
1540 LET B=10
1550 LET A=INT (RND*6)+1
1560 LET GAS=500
1570 IF A<=4 THEN PRINT AT 4,26;
"SECA"
1580 IF A>4 AND A<=7 THEN PRINT
AT 4,26;"UMIDA"
1590 IF A>7 THEN PRINT AT 4,25;"
MOLHADA"
1600 IF A>4 AND A<=7 THEN LET VM
AX=VMAX*.9
1610 LET MA1=0
1620 IF A>7 THEN LET VMAX=VMAX*.
8
```

→

```
1630 LET O=0
1640 LET POS=1
1650 PRINT AT 1,22;"■"
1660 LET AVE=0
1670 LET VELOC=AVE
1680 LET ST=POS
1690 LET MA1=0
1700 INPUT VEMA
1710 LET POS=POS+1
1720 IF POS=P THEN LET VMAX=VMAX
*.6
1730 IF POS=11 OR POS=13 OR POS=
18 OR POS=20 OR POS=25 OR POS=27
 OR POS=33 OR POS=38 OR POS=40 O
R POS=44 OR POS=46 THEN GOSUB 26
10
1740 LET VE=INT (VEMA/10)+(VEMA<
10)
1750 LET MA=ABS (VEMA)-INT (ABS
(VEMA)/10)*10
1760 LET G=ABS (VE)
1770 IF MA=1 AND G>50 THEN GOSUB
 2700
1780 IF MA=2 AND G>40 THEN GOSUB
 2700
1790 IF MA=3 AND G>30 THEN GOSUB
 2700
1800 IF MA=4 AND G>20 THEN GOSUB
 2700
1810 IF MA=5 AND G>10 THEN GOSUB
 2700
1820 IF MA=0 OR MA>5 THEN GOTO 8
70
1830 IF MA1=0 AND MA>1 THEN GOTO
 1850
1840 GOTO 1950
1850 FOR J=1 TO 5
1860 PRINT AT 7,25;"COMO E■"
1870 PRINT AT 8,25;"SAIR"
1880 PRINT AT 9,25;"EM ";MA;" ?"
1890 FOR I=7 TO 9
1900 PRINT AT I,25;"        "
1910 NEXT I
1920 NEXT J
1930 LET POS=POS-1
1940 GOTO 1690
1950 LET AVE=AVE+VE
1960 LET MA1=MA
1970 LET O=O+AVE
1980 LET VELOC=VELOC+VE
1990 IF VELOC<0 THEN LET VELOC=0
2000 IF VELOC>VMAX THEN GOTO 253
0
2010 IF B<=5 AND VELOC>VMAX*.6 T
HEN GOSUB 2640
2020 IF D=1 AND VELOC>VMAX*.8 TH
EN GOSUB 2640
2030 LET VEME=O/(POS-1)
2040 LET MA1=1
2050 LET ROTA=VELOC/MA*100
2060 IF ROTA>9000 THEN GOTO 2400
2070 LET TEMPO=(POS-1)/VELOC*10
2080 LET ST=ST+TEMPO
2090 IF VE>=0 THEN LET GAS=GAS-R
OTA/250
2100 IF GAS<=0 THEN GOTO 2380
2110 LET LI=INT (A(POS)/100)
2120 LET CO=A(POS)-INT (A(POS)/1
00)*100
2130 PRINT AT LI,CO;"■"
2140 PRINT AT 16,7;"    ";AT 16,7
;VELOC
2150 IF POS>2 AND VELOC<60 THEN
GOSUB 2330
2160 PRINT AT 17,7;INT (VEME)
2170 PRINT AT 18,7;INT (ST*10)/1
0
2180 PRINT AT 16,22;INT (ROTA)
2190 PRINT AT 17,22;MA
2200 PRINT AT 18,22;"    ";AT 18,
22;INT (GAS)
2210 IF POS=52 THEN GOTO 2760
2220 IF POS=P THEN LET VMAX=VMAX
/.6
2230 IF POS=10 OR POS=12 OR POS=
17 OR POS=19 OR POS=24 OR POS=26
 OR POS=32 OR POS=37 OR POS=39 O
R POS=43 OR POS=45 THEN GOSUB 24
80
2240 IF D=1 THEN LET VMAX=VMAX/.
3
2250 IF D=1 THEN LET D=0
2260 REM SORTEIO COND/PISTA
2270 LET B=INT (RND*100)
2280 IF B<=5 THEN PRINT AT 14,26
;"OLEO"
2290 IF B<=5 THEN LET P=POS+2
2300 IF B>5 THEN LET P=0
2310 IF B>5 THEN PRINT AT 14,26;
"-OK-"
2320 GOTO 1700
2330 FOR E=1 TO 6
2340 PRINT AT 8,25;"TARTA";AT 9,
25;"RUGA."
2350 PRINT AT 8,25;"     ";AT 9,
25;"     "
2360 NEXT E
2370 RETURN
2380 PRINT AT 7,25;"ACABOU";AT 8
,27;"A";AT 9,25;"GASOL.";AT 10,2
5;"E■ O FIM"
2390 STOP
2400 PRINT AT 7,25;"FUNDIU";AT 8
,27;"O";AT 9,25;"MOTOR";AT 10,25
;"E■ O FIM"
2410 FOR E=1 TO 5
2420 FOR W=1 TO 3
2430 PRINT AT 16,22;INT (ROTA)
2440 NEXT W
2450 PRINT AT 16,22;"     "
2460 NEXT E
2470 STOP
2480 FOR E=1 TO 6
2490 PRINT AT 8,25;"CUIDA";AT 9,
27;"DO"
2500 PRINT AT 8,25;"     ";AT 9,
27;"  "
2510 NEXT E
2520 RETURN
2530 FOR E=1 TO 6
2540 PRINT AT 7,25;"EXCESSO";AT
8,25;"  DE";AT 9,25;"VELOC."
2550 PRINT AT 7,25;"       ";AT
8,25;"       ";AT 9,25;"       "

2560 NEXT E
2570 PRINT AT 6,25;"E■ O FIM"
2580 PRINT AT 7,25;"TE VEJO";AT
8,28;"NO"
2590 PRINT AT 9,25;"FUNERAL"
2600 STOP
2610 LET VMAX=VMAX*.3
2620 LET D=1
2630 RETURN
2640 LET ST=ST+.01
2650 FOR E=1 TO 6
2660 PRINT AT 7,25;"DERRA";AT 8,
25;"PAGEM"
2670 PRINT AT 7,25;"     ";AT 8,
25;"     "
2680 NEXT E
2690 RETURN
2700 FOR E=1 TO 6
2710 PRINT AT 7,25;"INFORMA";AT
8,25;"  CAO";AT 9,25;"ERRADA."
2720 PRINT AT 7,25;"       ";AT
8,25;"       ";AT 9,25;"       "

2730 NEXT E
2740 LET POS=POS-1
2750 GOTO 1700
2760 LET ES=VEME*GAS/ST
2770 PAUSE 180
2780 CLS
2790 PRINT TAB 8;"P A R A B E N
S"
2800 PRINT
2810 PRINT "     VOCE CONSEGUIU T
ERMINAR"
2820 PRINT ,,,,TAB 13;"■"

2830 PRINT AT 10,6;"SEU ESCORE F
OI DE:"
2840 PRINT
2850 PRINT "        ";INT (ES);" PO
NTOS"
2860 STOP
2870 SAVE "F■"
2880 RUN
```

RESULTADO FINAL

Alcançando-se o ponto de chegada, é calculado um escore, cuja fórmula é:

$$P = \frac{v \times (500 - c)}{t}$$

onde:
t = tempo final de percurso
v = velocidade média
c = combustível utilizado
que pode ser utilizado de parâmetro comparativo entre vários "pilotos".

RESTRIÇÕES

Sem restrições o jogo não teria graça. Assim, existem certas limitações, que são:

a) para cada marcha, o acréscimo ou decréscimo máximo permitido é de:

marcha	1ª	2ª	3ª	4ª	5ª
km/h	50	40	30	20	10

b) o motor admite uma rotação máxima de 9.000 rpm. Acima deste valor, ele funde;

c) o carro parte com uma disponibilidade de 500 litros de gasolina especial, e seu consumo é calculado com base na rotação, que por sua vez depende da marcha e da velocidade. O combustível é limitado e não é permitido o reabastecimento no meio do circuito;

d) a velocidade máxima, nas retas (pista em boas condições e seca) é de 450 km/h. Entretanto:
— nas curvas este limite fica reduzido em 30 por cento;
— velocidade próxima do limite pode causar derrapagem ocasional e conseqüentemente, o acréscimo de dois segundos no tempo utilizado até então;
— à cada posição é simulado a presença ou não de uma poça de óleo, duas posições à frente (5 por cento de probabilidade de existência). A existência reduz o limite em 60 por cento;
— no início da corrida é simulado as condições do tempo e da pista: 50 por cento de probabilidade de estar seca; 30 por cento de probabilidade de estar úmida; e 20 por cento de probabilidade de estar encharcada.

Nas duas últimas condições, o limite fica reduzido em 10 e 20 por cento respectivamente;

e) excesso de velocidade, falta de combustível e o excesso de rotação do motor eliminam o piloto da corrida e do jogo.

O PROGRAMA

Todo o programa é escrito em BASIC, facilitando a sua compreensão, o que não implica em lentidão. Para os que desejam envenená-lo as indicações da tabela I são necessárias. A tabela II mostra algumas sugestões para incrementar o jogo.

Tabela I

Indicações sobre o programa

linhas	conteúdo
1 a 400	explicações iniciais
430 a 850	determinação de valores, que permitam o cálculo das coordenadas (modo PRINT), das 52 posições do circuito (vide 2130 e 2120)
870 a 1490	desenho do circuito (pista de corrida)
1500 a 1690	estabelecimento das condições iniciais
1700	variável de entrada VEMA = VE + MA (velocidade + marcha) vide 1740 a 1750
1950 a 2320	cálculo das condições parciais do jogo
1550	simulação das condições iniciais do tempo e da pista
2270	simulação da presença ou não de óleo na pista
1500	determinação da velocidade máxima
1560	determinação da capacidade do tanque de combustível
2060	determinação da rotação máxima do motor
2480 a 2750	avisos diversos quando da ocorrência de certas situações
2760	cálculo do número de pontos alcançados ao final do jogo
2870	para salvar o programa com o "label", "F1"

Tabela II

Sugestões para melhorar o programa

dar várias voltas no circuito, permitindo o reabastecimento com tempo aleatoriamente calculado;
apagar o sinal da posição anteriormente ocupada;
mostrar, como valor parcial, a temperatura do líquido de refrigeração e a pressão do óleo e simular a ocorrência de problemas neste sistema, e parada por quebra ou incêndio;
simular a necessidade de parada nos boxes para troca de pneus ou revisão.

○

POR DENTRO DO APPLE

FORMATAÇÃO EM APPLESOFT

José Eduardo Moreira
Wilson José Tucci

A maioria das pessoas que programam em FORTRAN, acham o comando FORMAT uma "chatice", muitas vezes se esquecendo do total controle sobre o formato de entrada e saída dos dados do programa, que este comando dá. Obviamente, o comando PRINT do BASIC, dispensando especificações de formato, é muito mais prático. Mas, muitas vezes queremos controlar a formatação dos números a serem impressos (ou visualizados), por isso, alguns BASICs como o Microsoft BASIC, permitem a opção PRINT USING, em que o formato dos números pode ser especificado.

No Applesoft BASIC não há, nenhum comando, seja obrigatório ou opcional, para formatar números, todo número é mostrado com todos os seus dígitos significativos. Além disso não há controle sobre a posição do ponto decimal (chamado de tabulação decimal). Ao se imprimir uma tabela, um bom resultado visual é obtido quando os números são impressos com os pontos decimais alinhados um em cima do outro.

Só para se ter uma idéia de como o Applesoft está longe de conseguir isso, rode o seguinte "programinha":

```
10  FOR J = 400 TO  - 400
 STEP  - 100:
PRINT J / 7: NEXT : END
```

Podemos construir uma sub-rotina que ajude a "consertar" isso. Precisamos usar o comando TAB(x) para iniciar a impressão do número na coluna certa. Obviamente *x* é função do número de dígitos significativos antes do ponto que o número tem. Se um número tem muitos dígitos antes do ponto, devemos começar a imprimir antes. Se ele tem poucos dígitos, devemos começar a imprimir depois. Lembrando que, em Applesoft, uma operação lógica retorna 1 se for *verdadeira* e Ø se for *falsa.* Assim podemos escrever a seguinte sub-rotina, que calcula a tabulação T para um certo número N.

```
1000 T = 10 - (N >  = 100
00) - (N >  =
1000) - (N >  = 100) - (N
 >  = 10
) - (N >  = 0)
1010  RETURN
```

Para testar, vamos rodar o seguinte programa:

```
10  FOR J = 400 TO  - 400
 STEP  - 100:
N = J / 7: GOSUB 1000: PR
INT  TAB( T)N: NEXT : END
```

Ótimo, não?! Bom, pelo menos para os números positivos. Podemos modificar a sub-rotina para tratar também números negativos:

```
1000 I = 1 - (2 * (N < 0)
)
1010 T = 10 - ((N * I) >
 = 10000) - (
(N * I) >  = 1000) - ((N
* I) >  = 10) - ((N * I) >
  = 0) - (N < 0)
1020  RETURN
```

O comando I = 1 — (2 * (n 0)) deixa I valendo 1 sen N 0 e valendo — 1 caso contrário, assim na linha 1010 comparamos sempre o módulo de N (podiamos também ter usado a função ABS). Se o número é negativo, subtraímos ainda mais um espaço para dar lugar ao sinal negativo.

Basta agora correr o programa e você verá que a tabulação decimal está totalmente implementada. O próximo passo é limitar o número de casas depois do ponto. Se quisermos ter apenas duas casas depois da vírgula, basta acrescentar uma linha à sub-rotina:

```
1000 N =  INT (N * 100 +
.5) / 100
1010 I = 1 - (2 * (N < 0)
)
1020 T = 10 - ((N * I) >
 = 10000) - (
(N * I) >  = 1000) - ((N
* I) >  = 100) - ((N * I)
>  = 10) - ((N * I) >
 = 0) - (N < 0)
1030  RETURN
```

Em geral, para "f" casas depois do ponto, escrevemos:

```
1000 N =  INT (N * 10 ^ F
 + .5) / 10 ^
F
1010 I = 1 - (2 * (N < 0)
)
1020 T = 10 - ((N * I) >
 = 10000) - (
(N * I) >  = 1000) - ((N
* I) >  = 100) - ((N * I)
>  = 10) - ((N * I) >
 = 0) - (N < 0)
1030  RETURN
```

Essa rotina é ótima para imprimir números, mas se quisermos imprimir valores em dinheiro, pode apenas colocar "$" antes do número modificando a linha 10 para:

```
10  FOR J = 400 TO  - 400
 STEP  - 100:
N = J / 7: GOSUB 1000: PR
INT  TAB( T)"$"N: NEXT : E
ND
```

então você irá observar alguns pequenos problemas:

1. se o número é negativo, o sinal negativo é impresso depois do cifrão ($ — 12.37);
2. se o número é menor que 1, não será impresso um 0 antes do ponto decimal;
3. se o número tem apenas 1 ou nenhuma casa decimal, não serão impressos zeros para completar.

Para resolver estes problemas iremos transformar o número N numa cadeia de caracteres (N$) e então manipulamos essa cadeia.

Para começar, vamos fazer uma sub-rotina que arredonda N e o transforma numa cadeia:

```
1000 N =  INT (N * 100 +
.5) / 100
1010 N$ =  STR$ (N)
1020  RETURN
```

Modificamos a linha 10 para:

```
10  FOR J = 400 TO  - 400
 STEP  - 100:
N = J / 7: GOSUB 1000: PR
INT N$: NEXT : END
```

e rodamos o programa.

O que fizemos não foi suficiente para colocar 0s complementares. Podemos dar um jeito fácil nisso:

```
1000 N =  INT (N * 100 +
.5) / 100 + .
001
1010 N$ =  STR$ (N)
1020 N$ = "$" +  LEFT$ (N
$, LEN (N$) -
1)
1030 RETURN
```

Agora, forçamos o número a ter uma terceira casa decimal, que jogamos fora, na linha 20, pegando todos os elementos da cadeia menos o último.

Modifiquemos agora a sub-rotina para tratar números menores que 1 e números negativos:

```
1000 N$ = "$": IF  ABS (N
) < 1 THEN N$
 = N$ + "0": IF N = 0 THE
N N$ = "
0.00": GOTO 1050
1010  IF N < 0 THEN N$ =
"-" + N$
1020 N =  INT ( ABS (N) *
 100 + .5) /
100 + .001
1030 N$ = N$ +  STR$ (N)
1040 N$ =  LEFT$ (N$, LEN
 (N$) - 1)
1050  RETURN
```

Para acertar de novo a posição do ponto, temos que incluir espaços suficientes na frente do número. A linha 1050 cuida disso:

```
1000 N$ = "$": IF  ABS (N
) < 1 THEN N$
 = N$ + "0": IF N = 0 THE
N N$ = "
0.00": GOTO 1050
1010  IF N < 0 THEN N$ =
"-" + N$
1020 N =  INT ( ABS (N) *
 100 + .5) /
100 + .001
1030 N$ = N$ +  STR$ (N)
1040 N$ =  LEFT$ (N$, LEN
(N$) - 1)
1050  IF  LEN (N$) < 10 T
HEN N$ = " " + N$: GOTO 10
50
1060  RETURN
```

E a sub-rotina está pronta!

Agora você tem duas pequenas sub-rotinas para formatar números. Uma inteiramente numérica e outra que inclui processamento de caracteres.

Boa aplicação!

Cadastro

APPLE

Mário Folli

Este é um programa multi-cadastral, desenvolvido em um micro computador Apple com 48K de memória RAM e uma unidade de discos.

É dirigido principalmente à utilização pessoal, com a finalidade de manter pequenos registros de diversas modalidades, tais como: movimentação da conta bancária, dados de amigos e parentes (para remessa de cartões de Natal), etc.

Seu manuseio é extremamente simples, uma vez que todas as funções disponíveis são acessadas por menus, bastando o toque de apenas uma tecla para obter a função escolhida.

Caso seja cometido algum erro na sua utilização, será devolvido o código deste erro na tela, retornando, em seguida, para o menu principal, o que possibilita ao usuário contornar o problema sem a interrupção do programa, o que causaria a perda de todos os registros da memória se o programa fosse rodado novamente.

Cadastro está dividido em 50 registros com 7 setores cada um, podendo guardar informações de até 23 caracteres por setor.

Possui rotinas de edição, deleção, gravação e leitura, impressão, listagem de registros, procura por substrings e procura por número de registros.

Para definir os 7 setores conforme suas necessidades, basta alterar a linha 8000 (instrução DATA) para as novas especificações e gravar novamente o programa, agora com um nome diferente de "CADASTRO". Porém, com a utilização do programa estruturador, esta tarefa fica ainda mais simples e organizada, além de mostrar na tela todos os setores alocados, conforme serão apresentados no programa de cadastro.

Principais Rotinas

2000 a 2090 — Efetua a listagem dos registros
3000 a 3430 — Efetua procura de registros
4080 a 4190 — Rotina de gravação de registros (tipo seqüencial)
4500 a 4590 — Leitura de registros do disquete
4900 a 4990 — Imprime registros
5000 a 5084 — Edição de registros e setores
5100 a 5170 — Oferece opções a cada registro completado
5200 a 5270 — Cancela registros da memória, rearranjando os demais
6000 a 6040 — Acusa erros de operação, caso sejam cometidos.

Programa Estruturador

```
1  REM  ************************
********
2  REM  * PROGRAMA ESTRUT
URADOR *
3  REM  *   AUTOR: MARIO
FOLLI  *
4  REM  *    NOVEMBRO DE
1984   *
5  REM  ************************
********
10  DIM D$(7): TEXT : HOM
E : PRINT : PRINT
20  PRINT "  ESTE E' O PR
OGRAMA ESTRUT
URADOR QUE   DEFINE OS SE
TORES DE
 CADASTRAMENTO."
30  PRINT "  O COMPRIMENT
O MAXIMO QUE
CADA SETOR   PODERA' TER,
 ESTA' L
IMITADO A DEZ(10)   CARAC
TERES."
40  PRINT "  E' INTERESSA
NTE QUE VOCE
FINALIZE O   NOME DO SETO
R COM DO
IS PONTOS(:), MAS   NAO E
' UMA OB
RIGATORIEDADE."
50  PRINT : PRINT : PRINT
 : PRINT "<RE
TURN>: CONTINUA": PRINT "
<ESCAPE>
: CANCELAR"
60  VTAB 14: HTAB 20: GET
 A$
62 A =  ASC (A$): IF A =
13 THEN 70
64  IF A = 27 THEN  HOME
: END
66  GOSUB 500: GOTO 60
70  HOME : FOR V = 2 TO 1
4 STEP 2
72  VTAB V: HTAB 1: PRINT
 V - V / 2;")
"
75  VTAB V: HTAB 15: PRIN
T "''''''''''
'''''''''''''''"
80  VTAB V: HTAB 4: INVER
SE : PRINT "
          ": NORMAL : NEXT

90  VTAB 18: HTAB 4: PRIN
T "ENTRAR COM
 O TITULO DE CADA SETOR"
100  HTAB 10: PRINT "FINA
LIZE COM <RET
URN>":V = 0:N = 1
110 V = V + 2:D$ = "":TX
= 4
120  VTAB V: HTAB TX: GET
 X$:X =  ASC
(X$)
130  IF X = 21 THEN X$ =
" "
140  IF X = 13 THEN 200
150  IF X = 8 THEN 190
160  IF TX = 14 THEN  GOS
UB 500: GOTO
120
170  VTAB V: HTAB TX: INV
ERSE : PRINT
```

```
X$: NORMAL
180 D$ = D$ + X$:TX = TX
+ 1: GOTO 120

190  IF TX <  = 5 THEN  V
TAB V: HTAB 4
: INVERSE : PRINT " ": GO
SUB 500:
V = V - 2: NORMAL : GOTO
110
195 TX = TX - 1:D$ =  LEF
T$ (D$,TX - 4
): VTAB V: HTAB TX: INVER
SE : PRINT " ": GOTO 120
200 D$(N) = D$:N = N + 1
210  IF V < 14 THEN 110
220  NORMAL : POKE 34,15:
 HOME : GOSUB
500: GOSUB 500
230  VTAB 18: PRINT "<RET
URN>: CONTINU
A": PRINT "<ESCAPE>: REDE
FINE"
240  VTAB 18: HTAB 20: GE
T A$:A =  ASC
(A$)
250  IF A = 27 THEN  POKE
 34,0: HOME :
 GOTO 70
260  IF A = 13 THEN 280
270  GOSUB 500: GOTO 240
280  HOME : VTAB 18
290  PRINT "ENTRE COM O N
OME DO SEU NO
VO PROGRAMA"
300  PRINT : PRINT "''''''
'''''''''''''
''''''''"

310  VTAB 20: INPUT "";F$

320  IF F$ = "" THEN F$ =
 "SEM NOME"
330  IF  LEN (F$) > 25 TH
EN  GOSUB 500
: GOTO 280
335  POKE 34,0: HOME
340  PRINT "  QUANDO O CU
RSOR RETORNAR
, VOCE DEVE   CONDUZI-LO
AO FINAL
```

```
 DE CADA INSTRUCAO   E PR
ESSIONAR
 <RETURN>."
350  PRINT "  SEU NOVO PR
OGRAMA SERA'
GRAVADO  NO   DISKETTE CO
M AS SUA
S DEFINICOES."
360  PRINT "  CONDUZINDO
O CURSOR ATE'
 O FINAL DA   ULTIMA LINH
A E <RET
URN>, SEU PROGRAMA   IRA'
 RODAR C
OM OS NOVOS SETORES."
365  PRINT : INVERSE : PR
INT "*USE A S
ETA => PARA CONDUZIR O CU
RSOR": NORMAL
370  FOR I = 1 TO 7
380  IF  LEN (D$(I)) < 10
 THEN D$(I) =
D$(I) + " ": GOTO 380
390  NEXT :A$ =  CHR$ (34
):B$ =  CHR$
(44)
400  VTAB 15: HTAB 2: PRI
NT "8000 DATA
 ";A$;D$(1);A$;B$;A$;D$(2
);A$;B$;
A$;D$(3);A$;B$;A$;D$(4);A
$;B$;A$;
D$(5);A$;B$;A$;D$(6);A$;B
$;A$;D$(
7);A$
410  VTAB 19: HTAB 2: PRI
NT "SAVE ";F$

420  VTAB 20: HTAB 2: PRI
NT "LOCK ";F$

430  VTAB 21: HTAB 2: PRI
NT "RUN"
440  VTAB 13: PRINT
450  PRINT  CHR$ (4);"LOA
D CADASTRO"
460  END
500  FOR I = 1 TO 20:S =
 PEEK ( - 163
36): NEXT : RETURN
```

```
0  REM  ****************
*******
1  REM  * CADASTRO: OUTUB
RO/84 *
2  REM  *   POR MARIO FOL
LI    *
3  REM  ****************
*******
4  DIM D$(700)
5 L$ = "================
=============
=========="
7  IF NR > 0 THEN 50
10  TEXT : HOME : PRINT :
 PRINT L$: VTAB 22: PRINT
L$
20  VTAB 3: INVERSE : PRI
NT " **** APP
LE II ###### OUTUBRO/1984
 **** ":
 VTAB 21: PRINT " **** AP
PLE II #
##### OUTUBRO/1984 **** "
: NORMAL

30  SPEED= 100: VTAB 7: H
TAB 8: PRINT
"PROGRAMA DE CADASTRAMENT
O": VTAB
8: HTAB 8: PRINT "*******
*******
**********": SPEED= 255:
VTAB 11:
 HTAB 16: PRINT "ESCRITO"
: VTAB 1
2: HTAB 18: PRINT "POR":
VTAB 13:
 HTAB 14: PRINT "MARIO FO
LLI"
40  FOR I = 1 TO 2000: NE
XT : GOSUB 50
90: GOSUB 5090: ONERR  GO
TO 6000
50  POKE 34,0: HOME :X =
0:Y = 0:NHR =
1
60  PRINT : PRINT L$: VTA
B 5: PRINT L$
```

```
70  VTAB 3: PRINT "  REGI
STROS FEITOS
   REGISTROS LIVRES": VTA
B 4: HTAB 9: PRINT "=  =";
: HTAB 29: PRINT "=  =
"
80  VTAB 4: HTAB 10: PRIN
T NR;: HTAB 3
0: PRINT 50 - NR
90  IF NR < 10 THEN  VTAB
 4: HTAB 10: PRINT 0;NR
100  IF NR > 40 THEN  VTA
B 4: HTAB 30:
 PRINT 0;50 - NR
110  VTAB 6: HTAB 13: INV
ERSE : PRINT
" MENU PRINCIPAL ": NORMA
L : VTAB
8: PRINT  TAB( 6)"INICIAR
 NOVO CA
DASTRO......(1)": VTAB 10
: PRINT
 TAB( 6)"CONTINUAR A CADA
STRAR...
...(2)"
120  VTAB 12: PRINT  TAB(
 6)"LISTAR OS
 REGISTROS........(3)": V
TAB 14: PRINT  TAB( 6)"PRO
CURAR REGISTROS.......
..(4)": VTAB 16: PRINT  T
AB( 6)"A
CESSAR PERIFERICOS.......
.(5)": VTAB 18: PRINT  TAB
( 6)"ENCERRAR O PROCES
SO........(6)"
130  VTAB 21: HTAB 5: INV
ERSE : PRINT
" ENTRE COM O NUMERO SELE
CIONADO
": NORMAL
140  VTAB 22: PRINT L$
150  VTAB 13: HTAB 37: GE
T A$:A =  VAL
(A$): IF A < 1 OR A > 6 T
HEN  GOSUB 5090: GOTO 150
160  ON A GOSUB 1000,1500
,2000,3000,40
00,7000
170  GOTO 50
```

```
1000  HOME : VTAB 3: PRIN
T "**********
***** ATENCAO ***********
*****": GOSUB 5090: GOSUB
5090: GOSUB 5090: SPEED= 1
70
1010  PRINT " - INICIANDO
 UM NOVO CADA
STRO,TODOS OS  REGISTROS
QUE POR
VENTURA ESTIVEREM NA  MEM
ORIA DO
COMPUTADOR, SERAO PERDIDO
S.": SPEED= 255
1020  VTAB 16: PRINT "'RE
TURN'= INICIA
R CADASTRO": VTAB 17: PRI
NT "'M'=
 MENU PRINCIPAL"
1030  VTAB 16: HTAB 30: G
ET A$
1040  IF A$ = "M" THEN 50

1050  IF A$ <  > CHR$ (1
3) THEN  GOSUB 5090: GOTO
1030
1070 NR = 1:NHR = 1: ONER
R  GOTO 6000
1090  POKE 34,0: HOME : H
TAB 16: INVERSE : PRINT "
REGISTRO #";NR: NORMAL : G
OSUB 5300
1100  FOR II = 4 TO 16 ST
EP 2
1110 V = II:L = K:CAD = 1
: GOSUB 5060
1120  FOR AP = TX TO 40:
VTAB II: PRINT " ";: NEXT

1130 D$(K) = D$:K = K + 1
: NEXT
1140 X = 1: PRINT  CHR$ (
7): GOSUB 510
0
1150  IF NR = 50 THEN 750
0
1160  RESTORE :NR = NR +
1: GOTO 1090
1500  IF NR = 0 THEN 1070
```

```
1510  GOTO 1150
2000  POKE 34,12: HOME :
POKE 34,0
2001  VTAB 14: PRINT L$
2002  VTAB 16: INPUT "INI
CIANDO EM QUA
L REGISTRO ? ";A$:A =  VA
L (A$)
2003  IF NR = 0 THEN 2080

2004  IF A < 1 OR A > NR
THEN 2002
2005 NS = A:T = NS * 7 -
7: HOME : GOSUB 5300
2010  VTAB 1: HTAB 16: IN
VERSE : PRINT
" REGISTRO #";NS: NORMAL

2020  FOR I = 4 TO 16 STE
P 2
2030  VTAB I: HTAB 15: PR
INT D$(T);"
                         "
2040 T = T + 1: NEXT
2050  GOSUB 5100: GOSUB 5
300:NS = NS +
1
2055  IF NR = 0 THEN  POK
E 34,0: HOME
: GOTO 2080
2060  IF NS <  = NR THEN
2010
2070  POKE 34,18: HOME
2080  VTAB 20: PRINT " 'R
EGISTROS ESGO
TADOS'": VTAB 22: PRINT "
PRESSION
E QUALQUER TECLA": GOSUB
5090: GOSUB 5090
2090  VTAB 22: HTAB 26: G
ET A$: RETURN

3000  POKE 34,14: HOME :
VTAB 16: PRINT L$:X = 0:NH
R = 0:CC = 0:CE = 0
3010  VTAB 18: PRINT "PRO
CURAR POR SUB
STRING..(A)": VTAB 19: PR
INT "PRO
```

```
CURAR POR REGISTROS..(B)"
: VTAB 2
0: PRINT "CANCELAR A PROC
URA.....
.(C)": VTAB 18: HTAB 30:
PRINT "(
SELECIONE)"
3020  VTAB 20: HTAB 35: G
ET A$
3030  IF A$ = "C" THEN  R
ETURN
3040  IF A$ = "B" THEN 33
00
3050  IF A$ (  ) "A" THEN
  GOSUB 5090:
 GOTO 3020
3060  POKE 34,0: HOME : G
OSUB 5300
3070 J = 1: FOR I = 4 TO
16 STEP 2
3080  VTAB I: PRINT J;")"
:J = J + 1: NEXT
3090  VTAB 18: PRINT "QUA
L SETOR PROCU
RAR?"
3100  VTAB 18: HTAB 23: G
ET A$:A =  VAL (A$)
3110  IF A ( 1 OR A ) 7 T
HEN  GOSUB 50
90: GOTO 3100
3120  RESTORE : FOR I = 1
 TO A: READ I
$: NEXT
3130  VTAB 20: HTAB 15: P
RINT "ENTRAR
COM O STRING": HTAB 15: P
RINT "PA
RA O SETOR ";I$
3150 V = 23:C$ = "":CAD =
 1: GOSUB 506
0:C$ = D$
3160  POKE 34,18: HOME :
VTAB 20: PRINT "EFETUANDO
A PROCURA": VTAB 21: PRINT

"AGUARDE UM MOMENTO"
3170  FOR IX = A - 1 TO 3
50 STEP 7
3180  IF C$ = D$(IX) THEN
 CE = 1: GOTO
3210
3185  NEXT : IF CE (  ) 1
 THEN 3500
3187  IF NHR = 0 THEN NH$
 = C$ + ":NAO
 REGISTRADO      "
3190  IF NHR (  ) 0 THEN
NH$ = "NAO HA
' MAIS REGISTROS"
3200  GOSUB 5090: GOSUB 5
090: VTAB 18:
 HTAB 1: PRINT NH$: GOSUB
 5100: GOTO 3000
3210 NS =  INT (IX / 7) +
 1:NHR = 1
3220 L = NS * 7 - 7
3240  POKE 35,2: HOME : P
OKE 35,24
3250  VTAB 1: HTAB 16: IN
VERSE : PRINT
" REGISTRO #";NS: NORMAL

3260  GOSUB 5300: FOR M =
 4 TO 16 STEP
2
3270  VTAB M: HTAB 15: PR
INT D$(L);"
                         ":L
 = L + 1
: NEXT
3280  VTAB 18: PRINT "(RE
GISTRO ENCONT
RADO)": GOSUB 5090
3290  GOSUB 5100: IF IX )
 350 THEN 318
7
3291 A = IX + 8
3292  IF CC = 0 THEN 3170

3293  GOTO 3500
3300  VTAB 22: HTAB 1: PR
INT "QUAL O N
UMERO DO REGISTRO ?": VTA
B 22: HTAB 29: INPUT "";NS

3310  POKE 34,0: HOME :NH
R = 1
3320  VTAB 1: HTAB 16: IN
VERSE : PRINT
" REGISTRO #";NS: NORMAL

3330  VTAB 21: PRINT "AGU
ARDE UM MOMEN
TO"
3340  IF NS ) NR THEN 342
0
3350 L = 0: FOR I = 1 TO
50
3360  IF NS = I THEN 3380

3370 L = L + 7: NEXT
3380  GOSUB 5300: FOR I =
 4 TO 16 STEP
2
3390  VTAB I: HTAB 15: PR
INT D$(L):L =
L + 1: NEXT
3395  IF Y = 1 THEN  RETU
RN
3400  VTAB 18: PRINT "(RE
GISTRO ENCONT
RADO)"
3410  GOSUB 5090: GOSUB 5
100: GOTO 300
0
3420  VTAB 23: PRINT "NAO
 HA' REGISTRO
": GOSUB 5090
3430  VTAB 23: HTAB 18: G
ET A$: GOTO 3
000
3500 W =  LEN (C$): FOR I
X = A - 1 TO
350 STEP 7
3510  IF C$ =  LEFT$ (D$(
IX),W) THEN C
C = 1: GOTO 3210
3520  NEXT : GOTO 3187
4000  HOME
4010  VTAB 2: INVERSE : P
RINT " ## GRA
VAR/RECUPERAR/APAGAR/IMPR
IMIR ##
": VTAB 16: HTAB 5: PRINT
 " ENTRE
```

```
 COM O NUMERO SELECIONADO
 ": NORMAL
4020  VTAB 5: PRINT  TAB(
 9)"VOLTAR AO
 MENU......(1)": VTAB 7:
PRINT  TAB( 9)"GRAVAR CADA
STRO.....(2)": VTAB 9:
 PRINT  TAB( 9)"CHAMAR CA
DASTRO..
...(3)": VTAB 11: PRINT
TAB( 9)"
APAGAR CADASTRO.....(4)":
 VTAB 13
: PRINT  TAB( 9)"IMPRIMIR
 REGISTR
OS..(5)"
4040  VTAB 9: HTAB 33: GE
T A$: PRINT  CHR$ (1):A =
 VAL (A$):G$ =  CHR$ (4)
4050  IF A = 2 THEN  GOSU
B 4080: GOSUB
4800: GOTO 4090
4055  IF A = 3 THEN  GOSU
B 4500: GOSUB
4800: GOTO 4510
4060  IF A = 4 THEN  GOSU
B 4600: GOSUB
4800: GOTO 4610
4064  IF A = 5 THEN 4900
4068  IF A = 1 THEN 50
4070  GOSUB 5090: GOTO 40
40
4080  VTAB 18: HTAB 16: I
NVERSE : PRINT " GRAVAR ":
 NORMAL : RETURN
4090  VTAB 20: HTAB 1: IN
PUT "NOME DO
CADASTRO=? ";F$
4092  IF F$ = "" THEN 409
0
4100  VTAB 22: HTAB 5: IN
VERSE : PRINT
" GRAVANDO...AGUARDE UM M
OMENTO "
: NORMAL
4110  PRINT G$;"OPEN";F$
4120  PRINT G$;"DELETE";F
$
4130  PRINT G$;"OPEN";F$
4140  PRINT G$;"WRITE";F$

4150  FOR I = 0 TO K
4160  PRINT D$(I): NEXT
4170  PRINT G$;"CLOSE";F$

4190  GOTO 50
4500  VTAB 18: HTAB 14: I
```

```
NVERSE : PRINT " RECUPERAR
 ": NORMAL : RETURN
4510  CLEAR : VTAB 20: HT
AB 1: INPUT "
NOME DO CADASTRO=? ";F$
4512  IF F$ = "" THEN 451
0
4520  VTAB 22: HTAB 6: IN
VERSE : PRINT
" LENDO...AGUARDE UM MOME
NTO ": NORMAL :G$ =  CHR$
(4)
4530  PRINT G$;"OPEN";F$
4540  PRINT G$;"READ";F$
4550  DIM D$(700):K = 0:
ONERR  GOTO 4
580
4560  INPUT D$(K)
4570 K = K + 1: GOTO 4560

4580 NR =  INT (K / 7): I
F NR = 1 THEN
NR = 0
4590 NS = NR:K = K - 1: G
OTO 5
4600  VTAB 18: HTAB 16: I
NVERSE : PRINT " APAGAR ":
 NORMAL : RETURN
4610  VTAB 20: HTAB 1: IN
PUT "NOME DO
CADASTRO=? ";F$
4620  IF F$ = "" THEN 461
0
4630  VTAB 22: HTAB 5: IN
VERSE : PRINT
" APAGANDO...AGUARDE UM M
OMENTO "
: NORMAL
4640  PRINT G$;"DELETE"F$
4650  GOTO 4000
4800  POKE 35,17: HOME
4810  PRINT G$;"CATALOG"
4820  POKE 35,24: RETURN

4900  HOME : VTAB 2: INPU
T "QUAL REGIS
TRO IMPRIMIR ? ";IMP$:IMP
 =  VAL
(IMP$)
4910  IF IMP < 1 OR IMP >
 NR THEN 4000

4920  HOME :IMP = IMP * 7
 - 7
4930  PRINT G$;"PR#1": PR
INT : PRINT
```

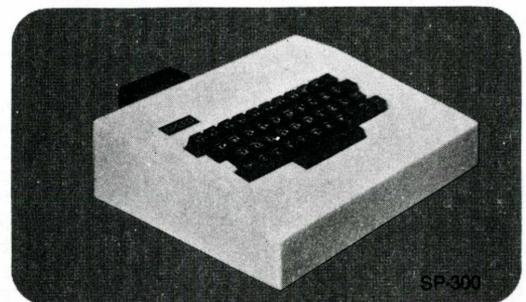
```
4940  RESTORE : FOR I = 1
 TO 7
4950  READ IMP$
4960  HTAB 2: PRINT IMP$;
: HTAB 13: PRINT D$(IMP)
4970 IMP = IMP + 1: PRINT
 : NEXT
4980  PRINT G$;"PR#0"
4990  GOTO 4000
5000  IF NHR = 0 THEN  VT
AB 22: HTAB 2
0: PRINT "(NADA EFETUADO)
": GOSUB
5090: GOSUB 5090: FOR I =
 1 TO 50
0: NEXT : RETURN
5005  POKE 34,18: HOME :
POKE 34,0: IF
X = 1 THEN NS = NR
5010 L = NS * 7 - 8:J = 1
:V = 0: FOR I
 = 4 TO 16 STEP 2: VTAB I
: HTAB 1
: PRINT J;")":J = J + 1:
NEXT
5020  VTAB 20: HTAB 1: PR
INT "QUAL SET
OR MODIFICAR ?"
5030  VTAB 20: HTAB 24: G
ET A$:A =  VAL (A$)
5040  IF A < 1 OR A > 7 T
HEN  GOSUB 50
90: GOTO 5030
5050 V = 2: FOR I = 1 TO
A:V = V + 2:L
 = L + 1: NEXT :CAD = 0
5060  FOR I = 15 TO 38: V
TAB V: HTAB I
: PRINT "'"; CHR$ (127):
NEXT
5070  FOR I = 38 TO 15 ST
EP  - 1: VTAB
V: HTAB I: PRINT  CHR$ (1
27);"'":
 NEXT : VTAB V: HTAB 15:
PRINT "'
"
5072 TX = 15:D$ = ""
5074  VTAB V: HTAB TX: GE
T X$:X =  ASC
(X$): IF X = 21 THEN X$ =
""
5076  IF X = 13 AND CAD =
 1 THEN  RETURN
5077  IF X = 13 AND CAD =
 0 THEN D$(L)
 = D$: FOR AP = TX TO 40:
```

```
VTAB V:
 PRINT " ";: NEXT : GOSUB
 5100: RETURN
5078  IF X = 8 THEN 5086
5079  IF TX = 40 THEN  GO
SUB 5090: GOTO 5074
5080  IF X = 44 THEN X$ =
 ","
5081  IF X = 58 THEN X$ =
 "_"
5082  VTAB V: HTAB TX: PR
INT X$
5084 D$ = D$ + X$:TX = TX
 + 1: GOTO 50
74
5085  GOSUB 5100: RETURN

5086  IF TX <  = 16 THEN
 VTAB V: HTAB
15: PRINT "'": GOSUB 5090
: GOTO 5
072
5088 TX = TX - 1:D$ =  LE
FT$ (D$,TX -
15): VTAB V: HTAB TX: PRI
NT "'": GOTO 5074
5090  FOR IJ = 1 TO 20:P
```

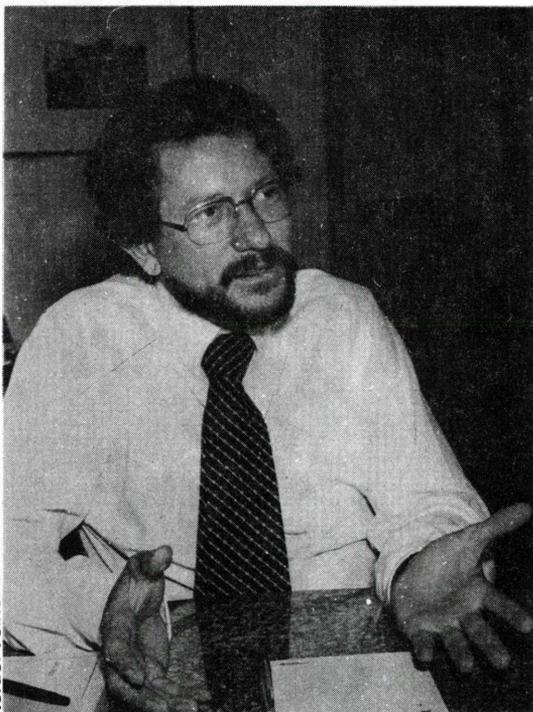
```
=  PEEK ( - 1
6336): NEXT : RETURN
5100  POKE 34,18
5110  VTAB 20: PRINT "'RE
TURN'= CONTIN
UAR  ": VTAB 21: PRINT "
'M'= MEN
U PRINCIPAL": VTAB 22: PR
INT "'A'
= APAGAR": VTAB 23: PRINT
 "'E'= E
DITAR"
5120  VTAB 20: HTAB 21: G
ET A$
5130  IF A$ =  CHR$ (13)
THEN  RETURN

5140  IF A$ = "M" THEN 50

5150  IF A$ = "A" THEN  G
OSUB 5200: RETURN
5160  IF A$ = "E" THEN  G
OSUB 5000: RETURN
5170  GOSUB 5090: GOTO 51
20
5200  IF NHR = 0 THEN  VT
AB 22: HTAB 2
```

```
0: PRINT "(NADA EFETUADO)
": GOSUB
5090: GOSUB 5090: FOR I =
 1 TO 50
0: NEXT : RETURN
5210  VTAB 22: HTAB 13: P
RINT "---> CO
NFIRME TECLANDO 'A'"; CHR
$ (7)
5220  VTAB 22: HTAB 39: G
ET A$
5222  VTAB 22: HTAB 13: P
RINT "
                         "
5224  IF A$ <  > "A" THEN
  RETURN
5225  IF K = 7 THEN NR =
0:K = 0: RETURN
5230  POKE 34,0: HOME
5240  VTAB 20: PRINT "AGU
ARDE UM MOMEN
TO...": IF X = 1 THEN NS
= NR
5250 L = NS * 7 - 7: FOR
I = L TO NR *
8
140
```

TENTE ESTA

```
1  ONERR  GOTO 5
5  INPUT "DURACAO (0 A 255, DUAS VEZES
)";C,D
10 A =  PEEK (38):B =  PEEK (39)
15  IF A = 1 THEN A = 48: IF B = 1 THEN
B = 0
20  IF A = 0 THEN A = 1: IF B = 48 THEN
B = 1
30  SOUND B,C TO A,D
40  GOTO 10
```

Ana Lúcia de Alcântara

Colaboraram Solange Aparecida de Menezes em São Paulo e Fátima França no Rio de Janeiro.

O USO DO COMPUTADOR A SOCIEDADE E O PROFISSIONAL

Israel Teixeira

"É necessário chegar-se à real concepção de informática para se pensar na conscientização do uso de seu instrumento — o computador". Henrique Costábile, presidente da SUCESU/SP.

O computador tem sido apresentado ao indivíduo como um instrumento solucionador dos problemas enfrentados pelos diversos setores da sociedade, sempre precedido de argumentos como: "o computador possibilita a redução de tarefas improdutivas", ou então: "o computador dá ao homem um maior espaço de tempo para que este se dedique mais ao seu lazer". Até que ponto estas considerações são verdadeiras é a questão principal. Muito pouco tem-se falado a respetio da conscientização do uso correto do computador, tanto pelo usuário final — o cidadão comum — quanto pelo profissional dos centros de processamento de dados.

A realidade da "sociedade de informação" já se configura no Brasil e os indivíduos se vêem à frente da influência do computador em todas as suas atividades. Diante deste fato, reacende-se um fator muito importante: as pessoas estão utilizando o computador, se defrontam com ele em todos os setores da sociedade e, finalmente, a sua vida está, de certa forma, sendo influenciada pelo uso cada vez mais constante do novo instrumento de trabalho. Neste ponto, um tópico tem levantado discussões e gerado alguns estudos. A principal preocupação existente em todos eles é quanto a conduta ético-moral do indivíduo frente à nova tecnologia.

Segundo Henrique Costábile, diretor do Citibank e presidente da SUCESU — Sociedade dos Usuários de Computadores e Equipamentos Subsidiados de São Paulo, para se chegar a uma conclusão a respeito desta problemática é necessário pensar-se a respeito da real concepção da informática. Para ele, é preciso encará-la como um meio e não como um fim, pois sua principal base de sustentação é a busca do desenvolvimento econômico do mundo e da felicidade da humanidade, eliminando e fazendo, de maneira mais rápida, as tarefas repetitivas. Porém, conforme acrescentou, a tecnologia de informática tem sido encarada sempre como um desafio técnico a ser vencido e geralmente é implantada para solucionar problemas, com o objetivo de tornar determinadas tarefas — que até então estavam sendo feitas manualmente — mais eficientes, diminuindo o seu custo operacional.

Esta tendência apontada por Costábile tem estado presente nos diversos setores da economia que passaram a adotar a nova tecnologia. Como exemplo deste fato, ele citou o caso da otimização do processo de controle da arrecadação do imposto de renda. "Na introdução da automatização no processo de arrecadação houve preocupação apenas com o aspecto técnico do sistema, ou seja, otimizou-se o processo, para torná-lo o mais perfeito possível", acrescentou. O sistema, segundo Costábile, é perfeito, porém foi implantado em cima da concepção, existente em todos os sistemas, que está baseada sempre no controle de alguma coisa, levando em conta apenas aspectos técnicos do problema. "Neste caso", disse, "não foi considerado o aspecto social, dentro da visão de que o sistema está prestando um serviço público ao cidadão com o objetivo de melhor serví-lo.

Este é um problema de 'âmbito cultural', que envolve a própria formação étnica do nosso povo. É uma marca presente nos diversos tipos de atividades. Primeiro faz-se as regras e depois se vê a prática. É o velho hábito de se fazer a norma antes da praxis", concluiu o presidente da SUCESU.

Privacidade de informações

O Brasil possue, segundo o SEI — Secretaria Especial de Informática, um parque de cerca de 23 mil computadores instalados, que estão sendo utilizados em diversos tipos de aplicações. Os Bancos de Dados destes computadores controlam vários tipos de informações e algumas destas são aquelas que dizem respeito à vida pessoal do indivíduo. José Roberto Faria Lima, diretor da Sisco Computadores, divide estas informações em setores: administrativas, aquelas que não devem ser de domínio público, as estatísticas, que dizem respeito mais diretamente à ética do profissional que lida com elas; e as pessoais, àquelas que exigem sigilo absoluto. Estas são as que envolvem mais diretamente o cidadão. E neste ponto, não existe nenhum tipo de legislação no Brasil que impeça os bancos de dados de intercambiarem estas informações, ou seja, que proibam a transferência de dados ou, como se denomina tecnicamente, o "transborder data flow", sem autorização do interessado.

Com outro ponto de vista, Frederico J.A. Buchman, diretor da Telemática, não acredita que haja condições de interação entre os bancos de dados. "No momento, as empresas têm como pontos importantes a segurança das informações e, se ocorre, em alguns casos, vazamento das mesmas, isto se deve mais à ética do profissional e da própria empresa, ou mesmo de sua organização interna". Segundo Buchman, tecnicamente não é possível acessar Bancos de Dados, principalmente por linha telefônica, pois existe legislação específica controlada pela Telebrás — Telecomunicações Brasileira, que proíbe este acesso. "Acessar Bancos de Dados, utilizando este método", afirmou, "só é possível através de linhas paralelas independentes e isto não existe".

Mas, conforme avaliação do professor Jorge da Cunha Pereira Filho em seu livro — Computadores para Usuários, volume I, as maiores perdas de dados que ocorrem nos sistemas de computação gira em torno daquelas causadas pelo que ele chama de "profissionais negligentes" (cerca de 84%), seguidas pelas causadas por vazamento de informações (16%), entre outras.

O problema é bastante complexo, pois a todo momento surgem novos Bancos de Dados que, na opinião do advogado Cândido Mendes, assessor jurídico da Assespro — Associação das Empresas de Processamento de Dados, nas mãos de um governo inescrupuloso, podem tornar-se armas perigosíssimas.

Na verdade, Tarcísio Cerqueira quer mostrar a necessidade existente no Brasil, de uma lei que haja de forma eficiente no controle de informações sobre um indivíduo que constem de alguns Bancos de Dados. Na França, Canadá, Suíça e outros países, já existem leis fiscais que regulamentam os Bancos de Dados e permitem aos cidadãos fazerem alterações nos dados de propriedade de determinadas empresas. Para ele, as informações contidas nestes Bancos deveriam ser controladas a fim de evitar que os cidadãos não fiquem subjugados, pois as leis são feitas para proteger as pessoas e evitar que estas se tornem habitantes de uma sociedade autoritária.

Proteção legal aos dados

"Não só através de Bancos de Dados a privacidade do indivíduo pode ser violada", explica o Juiz Pedro Luiz Ricardo Gagliardi, do Juizado de Menores do bairro da Penha em São Paulo. Exemplificando, Pedro Luiz disse que uma simples ficha de hotel pode ser uma fonte de informações perigosa se cair nas mãos de pessoas de má fé. Mas, segundo ele, o que realmente desencadeou a violação da privacidade como um probema foi a utilização de computadores.

Pedro Gagliardi defendeu a tese "Aspectos penais da violação da privacidade pela moderna tecnologia dos computadores", em 1981, e até hoje seu trabalho é vanguardista. Pouco se tem feito ou discutido a respeito no Brasil, embora o tema seja comum em outros países. Para o Juiz, a violação de informações sigilosas é inevitável: "antes a violação era feita pelo próprio governo na defesa do Estado. Mas hoje, conforme ilustrou, o que se pretende é criar uma lei de privacidade em defesa do cidadão que, por conseguinte, é objeto do Estado.

Israel Teixeira

Mas mesmo para aqueles que se sentem desprotegidos judicialmente, o jurista informa que já existem formas legais em vigor que podem defendê-los, mesmo não sendo dirigidas especificamente aos problemas causados pelos Bancos de Dados. Como exemplo, o advogado cita o artigo 153 da atual Constituição Federal que trata das garantias e direitos do indivíduo. Ele veta a violação de correspondências e ligações telefônicas e, neste caso, poderia se incluir a violação de informações pessoais. O que precisa ser feito, segundo Gagliardi, é apenas acrescentar um parágrafo a mais neste artigo, "de forma que tenhamos assegurada a possibilidade do cidadão consultar qualquer Banco de Dados, manual ou eletronicamente, para saber o que existe a seu respeito", ressaltou.

Como se proteger?

"As informações protegidas serão apenas referentes à pessoa e quais as informações que merecem proteção serão determinadas por uma lei federal, através de um carimbo de proteção. Dentro desta mesma lei haverá um parágrafo que dará direito ao cidadão ter acesso às suas próprias informações, podendo até corrigí-las em caso de erros. Essas correções serão feitas através da justiça que ouvirá as duas partes (o denunciante e o informante) e tomará a decisão", destacou o Juiz Gagliardi.

Já existem no Congresso Nacional alguns projetos de lei que prevêem a

Um estudo do código

Durante sua atuação como membro da APPD — Associação dos Profissionais de Processamento de Dados, Nilson Siqueira, atualmente Analista de Sistemas da Itaudata, elaborou um estudo com vista à estruturação de um código de conduta moral para sua categoria profissional. O principal objetivo deste código, conforme ressaltou, seria indicar as normas de conduta moral que deveriam inspirar as atividades dos profissionais, e regular suas relações com a classe, os poderes públicos e a sociedade. Assim sendo, baseando-se em códigos de ética de outras categorias profissionais, Nilson chegou a alguns tópicos que publicamos a seguir:

Segundo o estudo, são deveres do profissional de Processamento de Dados:

— Interessar-se pelo bem da sociedade e, com tal finalidade, contribuir com seus conhecimentos, capacidade e experiência; manter sigilo sobre o que souber em razão de suas atividades e de sua profissão; impedir que a privacidade dos cidadãos seja invadida;

— Conservar independência técnica na orientação dos serviços sob sua responsabilidade; recusar cargos, empregos ou funções, caso tenha consciência de que não dispõe de suficientes recursos técnicos para bem desempenhá-los; denunciar a prática de qualquer atividade profissional que contrarie o código; esclarecer ao leigo, a forma de atuação da área de informática.

São *deveres do Profissional de Processamento de Dados,* em relação a classe:

— Prestar apoio moral, intelectual e material às entidades de classe; desempenhar cargo diretivo nas entidades de classe, a não ser que circunstâncias especiais justifique a sua recusa; acatar as resoluções regularmente votadas pelas entidades de classe; não se aproveitar quando do desempenho de qualquer função diretiva em entidade representativa da classe, em benefício próprio.

Das disposições finais:

— A fiscalização e julgamento de infrações a este Código, decorrentes do exercício da atividade do profissional que atua em P.D., é inerente à associações de classe existentes, às quais caberá, por intermédio de seus *Conselhos de Ética,* aplicar as sanções cabíveis em cada caso.

Este estudo, porém, não foi levado avante pois ficou em teorias e no arquivo de Nilson.

proteção às informações concernentes à vida do cidadão. Entre estes, destacam-se o da deputada Cristina Tavares, com 32 artigos. Em alguns outros está previsto que o encarregado do Banco de Dados será o responsável pela guarda e coleta de informações. Prevê-se também nestes projetos, que a informação tenha proteção no Código Penal e a pessoa que se sinta ofendida poderá levar as provas à polícia e será aberto então, um inquérito policial.

Crimes contra o cidadão

Ultimamente a quebra de sigilo tem atingido proporções muito mais graves, chegando a prejudicar "o dono" das informações. Marco Antonio Laurindo, diretor da Informarket Consultoria, foi um dos atingidos com a informação indiscriminada de dados sigilosos. Recentemente ele recebeu um telefonema de uma empresa lhe cobrando a prestação atrasada de um apartamento que ele sequer tinha conhecimento. "Alguém teve acesso ao meu cadastro de informações, em algum lugar que eu não sei qual é, e não sei qual a intenção desta pessoa", disse ele.

Além deste existem outros casos que se tornam comuns hoje em dia. Um destes casos foi retratado por Nilson Siqueira, Analista do Sistemas da Itaúdata: a leitura de grafia, através de cartões perfurados. "Isto aconteceu em um grande shopping center de São Paulo. O sujeito, conhecedor da tecnologia, propagava a leitura da grafia do indivíduo, realizada segundo ele, através de cartões perfurados que forneciam os resultados quanto a personalidade do cidadão. Este caso, a meu ver é um abuso muito grande à ignorância da pessoa que não tem conhecimento de como funciona um computador".

Estas práticas criminosas para Murilo Pereira, diretor da Coopers & Lybrand, ligada à área de auditoria, tendem a se expandir com o passar dos anos e, no presente momento, não se tem como combatê-las judicialmente. Pode-se contar apenas com o bom senso das empresas preocupadas em proteger o cadastro de seus clientes e funcionários. Ele vê estes acontecimentos mais como um problema institucional que de processamento de dados.

O guardião das informações

Henrique Costábile, presidente da SUCESU/SP, acredita que a responsabilidade maior quanto a "guarda" dos dados dos sistemas de computação cabe ao profissional de P & D. Ele acredita que a discussão sobre um código de ética afeta muito mais ao Analista de Sistemas, pois é ele que elabora os sistemas. Estes, para Costábile, devem levar em conta principalmente a conservação das informações e a privacidade dos dados pessoais do cidadão.

Neste aspecto, Nilson Cerqueira, Analista de Sistemas, ressalta o fato de que, a maioria dos profissionais de sua categoria mantêm vínculo empregatício com a empresa para quem trabalham e que, por este motivo, estão sujeitos a pressões e à política adotada pela mesma. Nilson ressaltou que para o profissional seguir uma ética de conduta moral nesta área é necessário que haja proteção ao mesmo, dada pelas associações ou sindicatos. "O analista é aquele que sabe como fazer e não o que fazer. Para isto, a empresa já possue a estrutura. Existe no Brasil muito pouca conscientização quanto a ética ou postura profissional. Isto se deve principalmente ao não reconhecimento da profissão que é o primeiro caminho para se alcançar esta conscientização", concluiu.

José Maria Sobrinho, acredita também que o código de ética para os profissionais de processamento de dados deve levar em conta o fator "proteção ao Analista". Sobrinho crê que a elaboração de um código de ética deve especificar principalmente os limites de atuação e responsabilidade do profissional. Outro ponto destacado por ele é a não presença, nos currículos escolares — faculdades e universidades — de disciplinas que se direcionem à formação ético-moral e à conscientização dos futuros projetistas e processadores de sistemas de computação".

O advogado Tarcísio Cerqueira cita ainda outros aspectos. Um destes, é a proliferação de cursinhos livres de programação. Segundo ele, os grupos responsáveis pelo aparecimento destes cursos, além de não levarem em conta o perfil ético do profissional possuem pouco embasamento técnico e fazem uma formação em massa de mão-de-obra dita "capacitada", para exercer as diversas funções no ambiente de P.D.

Mas Tarcísio Cerqueira acredita que o profissional já está caminhando em direção à conscientização de seu papel dentro da sociedade. Conforme afirmou, há dez anos aproximadamente, quem trabalhava nos CPDs era considerado como um integrante do "Olimpo" empresarial, pois o grupo era quem ditava as normas de informatização deste ou daquele setor, ditando as regras. "Hoje", disse ele, "isso mudou". O sistema está se descentralizando e o analista de sistema já é uma figura mais comum dentro das empresas que estão se informatizando. Suas atribuições estão sendo divididas com os demais membros que atuam naquele setor. "Assim mesmo", garante, "informatizar uma firma, um setor é algo muito delicado e o analista, muitas vezes, tem de "fazer" um pouco de psicologia para vencer as resistências ao "novo".

Outro ponto que para o Juiz está mudando dentro dos CPDs é a mentalidade dos novos usuários. "As empresas que estão se informatizando" — explicou, "já começam a formar uma consciência da necessidade dos analistas deixarem seus "gabinetes" para sairem em busca de maiores dados sobre aquele universo que passará a fazer parte de seus programas. Ele disse ainda, que o técnico hoje tem muito mais vivência e autonomia para elaborar novos programas. "É preciso sair a campo", destacou.

Mesmo com este progresso, Padilha acredita que ainda há muito por fazer pois os profissionais de processamento de dados estão, apesar dos avanços de tecnologia nacional, trabalhando com máquinas e programas elaborados por outros povos, com culturas diferentes. "É preciso", frizou, "que os usuários sejam mais exigentes com aqueles computadores que vão utilizar e também com os softwares, a fim de que contribuam para um melhor e mais rápido aperfeiçoamento de uma política nacional de informática que hoje engatinha".

Israel Teixeira

Programas e proteção

Israel Teixeira

"A pirataria é um problema existente na maioria dos países e para combatê-la é necessário utilizar-se de processos judiciais". John Franklin Arce, diretor de informática do SENAI.

A discussão sobre a conscientização do uso do computador na sociedade não envolve apenas a privacidade de informações e segurança dos bancos de dados. Esta é uma questão muito ampla e atinge todos os aspectos que circundam este uso. Um destes aspectos refere-se às rotinas processuais do computador como por exemplo, o programa. Neste ponto tem surgido muitos debates que se voltam principalmente à má utilização e segurança dos programas — um fator bastante conhecido como pirataria —, cujos autores dos mesmos não têm como se protegem.

Conforme ressaltou John Franklin Arce, diretor de informática do SENAI, este é um problema igual em todos os demais países, inclusive com processos judiciais.

Nelson Costa, analista de sistema da BBS — Biblioteca Brasileira de Software vê o problema da pirataria dentro de um ponto de vista próprio. Para ele, a indústria de informática no Brasil começou errada, pois colocou-se a hardware à venda sem o mínimo suporte. Este processo, conforme afirmou, continua até hoje e tem colhido o desenvolvimento de programas nacionais.

Para Frederico Buchman, diretor da Telemática, este é um problema do código civil, pois ele acredita que copiar um programa nada mais é do que um "ato de roubo". Neste aspecto, a legislação deve ser bastante rigorosa, para punir indivíduos que se aproveitam do trabalho do companheiro, — no caso do CPD — que muitas vezes levou anos para desenvolver um sistema, ou programa, e passam para a frente com a maior facilidade". Ele fez questão de ressaltar: "a lei deve ser ampla para combater de forma sistemática o problema".

A respeito do problema, José Maria Sobrinho, diretor de informática da Sadia lembrou a Lei de Software que já circula no Congresso Nacional e que prevê a punição no caso de "copiagem".

Marco Antonio Laurindo, diretor da Informarket Consultoria, destacou dois aspectos que devem ser levados em conta no combate à pirataria: o técnico e o judicial. No plano técnico é muito difícil evitá-la devido a habilidade dos infratores. No plano judicial, conforme opinião do Luiz Pedro Luiz Ricardo Gagliardi será incluído um artigo no Código Penal na lei de privacidade e o infrator será condenado por furto:

— Mesmo não havendo ainda uma lei protetora, há leis similares que também podem ser aplicadas em caso de pirataria ou acesso indevido à informação. É o caso do artigo 155 da atual Constituição que diz: "subtrair para sí ou para outrem coisa alheia é considerado furto". Levando-se em conta que um programa é propriedade do fabricante pirateá-lo seria um furto. E a pena seria a mesma aplicada nos casos de furto de energia elétrica.

QUEBRA-CABEÇA

Sem FOR, sem IF e sem (!

Renato da Silva Oliveira

Numa das últimas cartas que recebi do Nabor, ele me relatou um fato bastante interessante, que acabou sugerindo o quebra-cabeça desta edição. Em janeiro deste ano, ele e a Dinorá deixaram a Inglaterra no Iate particular do Nabor.

Durante a viagem, o microcomputador que ele usava para calcular a rota caiu no chão e acabou sendo pisoteado acidentalmente pela Dinorá. Após o desastre, as instruções FOR e IF, e também o caractere (não estavam funcionando. Isso praticamente inutilizava todos os programas que deveriam ser introduzidos.

O Nabor conta que foi a própria Dinorá quem encontrou a solução para o problema.

A noite, após beber bastante suco de morangos silvestres do Ramarujan, a Dinorá mostrou a Nabor um programa de apenas sete linhas que preenchia totalmente a tela com asteriscos *, depois ele a limpava e começava tudo outra vez. Dinorá havia criado um laço potencialmente infinito no qual estava inserida implicitamente uma estrutura de decisão! O mais surpreendente é que o programa da Dinorá estava totalmente em BASIC e ela não usou sequer o PEEK ou o POKE!

Com esse programa, Dinorá mostrou que os micros tipo TK-85 podem tomar implicitamente decisões, mesmo sem as instruções FOR e IF, e mesmo sem o uso lógico dos parentesis.

O Nabor pode, então, introduzir seus programas, alterando-os ligeiramente de modo que só utilizassem as instruções que estavam funcionando.

O quebra-cabeça deste mês consiste em fazer um programa que produza o mesmo efeito que o da Dinorá:

1. Preencha a tela, posição por posição, com asteriscos.

2. Após preencher aos 704 posições regulares de impressão, limpe o vídeo.

3. Recomece tudo novamente a partir do item 1.

Claro que as instruções FOR e IF não devem ser utilizadas e nem os parêntesis!

EXPLORANDO O TK-2000

INTRODUÇÃO AOS INTERPRETADORES SEGUNDA PARTE

José Eduardo Moreira
Wilson José Tucci

No artigo anterior, estudamos um pouco da arquitetura da HP-33, bem como apresentamos rotinas de implementação para:

- apresentação da pilha operacional
- as 4 operações básicas (+ , —, *, /)
- o CLX
- o ENTER

Apresentaremos agora a rotina para entrada de dados.

As teclas para a entrada de dados são as teclas dos dígitos de Ø a 9, o ponto ".", a tecla EEX (que permite a entrada de expoente de 1Ø), e a tecla CHS que permite negativar o número que está sendo introduzido. Apertando qualquer outra tecla indicamos o fim do número e executamos a operação indicada pela tecla.

A seguinte sub-rotina faz a entrada de números:

```
20000  REM  ENTRADA DE DA
DOS
20010 B$ = ""
20020 L = 0
20030 E$ = ""
20040 ENTROU = 0
20050 P = 0
20060  VTAB 18: HTAB 10
20070  PRINT  SPC( 20): H
TAB 10
20080  GET A$:A =  ASC (A
$)
20090  IF A = 8 THEN 5000

20100  IF A$ = "C" AND B$
 = "" THEN 50
00
20110  IF A$ <  > "C" THE
N 20160
20120  IF  LEFT$ (B$,1) =
 "-" THEN B$ =  RIGHT$ (B$
, LEN (B$) - 1): GOTO 2014
0

20130 B$ = "-" + B$
20140  HTAB 10: PRINT  SP
C( 20);: HTAB
10: PRINT B$;
20150  GOTO 20080
20160  IF A$ = "E" AND B$
 = "" THEN B$
 = "1.": GOTO 20340
20170  IF A$ = "E" THEN 2
0340
20180  IF A = 13 AND B$ =
"" THEN 5000

20190  IF A <  > 13 THEN
20230
20200 NX =  VAL (B$) * 10
 ^  VAL (E$)
20210 ENTROU = 1
20220  GOTO 5000
20230  IF A < 46 OR A = 4
7 OR A > 57 THEN 5000
20240  IF A$ = "." THEN 2
0290
20250  IF L = 10 THEN  PR
INT  CHR$ (7)
;: GOTO 20080
20260 L = L + 1
20270 B$ = B$ + A$: HTAB
10: PRINT B$;

20280  GOTO 20080
20290  IF A$ = "." AND P
THEN  PRINT  CHR$ (7);: GO
TO 20080
20300  IF A$ = "." THEN P
 = 1: IF B$ =
"" THEN B$ = "0"
20310 B$ = B$ + A$
20320  HTAB 10: PRINT B$;

20330  GOTO 20080
20340  REM  ENTRADA DO EX
POENTE
```

```
20350  HTAB 25
20360 E$ = "00"
20370  HTAB 25: PRINT  SP
C( 3);: HTAB
25: PRINT E$; CHR$ (8);
20380  GET A$:A =  ASC (A
$)
20390  IF A$ = "E" THEN
PRINT  CHR$ (
7);: GOTO 20370
20400  IF A$ <  > "C" THE
N 20440
20410  IF  LEFT$ (E$,1) =
 "-" THEN E$ =  RIGHT$ (E$
,2): GOTO 20370
20420 E$ = "-" + E$
20430  GOTO 20370
20440  IF A <  > 13 THEN
20480
20450 NX =  VAL (B$) * 10
 ^  VAL (E$)
20460 ENTROU = 1
20470  GOTO 5000
20480  IF A < 46 OR A = 4
7 OR A > 57 THEN 5000
20490  IF A$ = "." THEN
PRINT  CHR$ (
7);: GOTO 20370
20500 E1$ = RIGTH$(E$,1)
+ A$
20510  IF  LEFT$ (E$,1) =
 "-" THEN E1$
 = "-" + E1$
20520 E$ = E1$
20530  GOTO 20370
```

A função EEX foi associada à tecla "E" e a função CHS à tecla "C".

Vamos aproveitar e implementar também algumas funções transcendentais da HP-33. Abaixo segue uma lista de funções, acompanhada da descrição da ação de cada uma delas.

SIN X : guarda o conteúdo de X em LX, calcula sen(X), guarda em X;
COS X: guarda o conteúdo de X em LX, calcula cos(X), guarda em X;
TAN X: guarda o conteúdo de X em LX, calcula tg(X), guarda em X;
LN X : guarda o conteúdo de X em LX, calcula ln(X), guarda em X;
LOG X: guarda o conteúdo de X em LX, calcula log(X), guarda em X;
Y : guarda o conteúdo de X em LX, calcula Y, guarda em X, baixa a pilha operacional.

As implementações dessas funções são apresentadas abaixo:

```
10400  REM  ^
10410 LX = X
10420 Y = Y ^ X
10430 Y = Z:Z = T
10440  RETURN
10500  REM  SIN
10510 LX = X
10520 X =  SIN (X)
10530  RETURN
10600  REM  COS
10610 LX = X
10620 X =  COS (X)
10630  RETURN
10700  REM  TAN
10710 LX = X
10720 X =  TAN (X)
10730  RETURN
10800  REM  LN
10810 LX = X
10820 X =  LOG (X)
10830  RETURN
10900  REM  LOG
10910 LX = X
10920 X =  LOG (X) /  LOG
 (10)
10930  RETURN
```

No próximo número apresentaremos a rotina da linha 5000, que reconhece o comando selecionado e dispara sua execução.

Curso de Assembly 6502

Veremos nesta lição algumas instruções do 6502 e faremos nossos primeiros programas.

Gustavo Egídio de Almeida

STA — Armazena o valor do acumulador na memória (tabela I).

TABELA I
Instruções STA

Código usado	Formato	Bytes usados
85	STA OPER	2
95	STA OPER, X	2
8D	STA OPER	3
9D	STA OPER, X	3
99	STA OPER, Y	3
81	STA (OPER,X)	2
91	STA (OPER), Y	2

As instruções STA fazem o papel inverso das instruções LDA. Observe o seguinte exemplo:

LDA $ 50	STA $ 50
1º	2º

No primeiro exemplo, como já é de nosso conhecimento, o acumulador é carregado com o conteúdo de endereço $ 50.

No segundo exemplo ocorre exatamente o contrário, ou seja, o endereço $ 50 é carregado com o valor do acumulador. A instrução STA (STORE) armazena em determinado endereço o valor do registrador que, dependendo do caso, pode ser A (acumulador), X (indexador X) ou Y (indexador Y).

STA OPER — Endereçamento direto (página zero) introduz o valor do acumulador no endereço definido pelo operador. Por exemplo: STA $ 50 — Introduz o valor do acumulador no endereço $ 50.

STA OPER, X — Endereçamento indexado por X (página zero). Reveja o funcionamento da instrução LDA OPER, X (página zero), lembrando que o sentido agora é o inverso.

STA $ 50, X — Supondo X = Ø8, coloca o valor do acumulador no endereço $ 58 ($ 50 + 'Ø8' = $ 58).

STA OPER — Endereçamento direto (Absoluto). Veja a instrução LDA OPER (Absoluto) da lição anterior.

STA $ 8ØØ — Coloca o valor do acumulador no endereço $ 8ØØØ.

STA OPER, X — Endereçamento indexado por X (Absoluto). Rever a instrução LDA, X (Absoluto).

Exemplo: STA $ Ø3ØØ, X — Supondo X = Ø5, coloca o valor do acumulador no endereço $ 3Ø5 ($ 3ØØ + 'Ø5' = $ 3Ø5).

STA OPER, Y — Endereçamento indexado por Y (Absoluto). Rever a instrução LDA OPER, Y (Absoluto).

Exemplo: STA $ Ø3ØØ, Y — Supondo Y = Ø7, coloca o valor do acumulador no endereço $ Ø3Ø7 ($ Ø3ØØ + 'Ø7' = $ 3Ø7).

STA (OPER,X) — Endereçamento indireto (indexado por X). Rever o funcionamento da instrução LDA (OPER,X). Veja o exemplo da figura 1.

Figura 1

STA ($10,X)
S 10 + 1071 = $ 17
+ 1
$ 18

Coloca o valor do acumulador no endereço ($ 1510)

Supondo: Conteúdo de X = 07
Conteúdo de 0017 = 10
Conteúdo de 0018 = 15

$ 0017 — #10
$ 0018 — # 15
} conteúdo dos endereços

Portanto, o endereço formado é $ 151Ø

STA (OPER, Y) — Endereçamento indireto (indexado por Y). Rever a instrução LDA (OPER,Y). Veja o exemplo da figura 2.

TABELA VI		
Código usado	**Formato**	**Bytes usados**
4C	JMP OPER	3
6C	JMP (OPER)	3

Dentro das instruções "JUMP", vimos inicialmente JSR, uma instrução que utiliza o salto para uma sub-rotina.

Ao encontrar a instrução RTS na sub-rotina, o programa volta com sua execução para o endereço logo após a instrução JSR.

JMP OPER — Endereçamento direto (Absoluto).

A instrução JMP, assim como a JSR, também realiza um salto para um determinado endereço, porém com a diferença de não haver mais volta ao programa principal que utilizou a instrução JMP, é como se existisse uma ligação imaginária unindo blocos de memória (figura 4).

Figura 4

```
0800 — BE 30 10    LDX $ 1030, X
0803 — B4 18       LDY $ 18, X
0805 — 9D 20 20    STA $ 2020, X
0808 — 4C 12 08    JMP $ 0812   ┐
080B — 00          BRK          │  Ligação imaginária
                                │  entre blocos
0812 — A9 1C       LDA # $ 1C  ┘
       8D 00 07    STA $ 0700
       8E 01 07    STX $ 0701
```

JMP — Endereçamento indireto (Absoluto). Essa instrução também realiza um salto, porém, o endereço para qual o "Jump" irá saltar não está especificado na instrução de modo direto e sim indireto (figura 5).

Figura 5

```
JMP (S 5010)
    + 1                    Dados { $ 5010 — # 60
  $ 5011                         { $ 5011 — # 10

$ 5010 — # 60 (parte baixa do endereço) }  endereço
$ 5011 — # 10 (parte alta do endereço)  }  formado:
                                           $ 1060
```

Portanto, no exemplo da figura 5, a instrução JMP irá saltar para o endereço $ 1060.

Sub-rotinas contidas na ROM

Existem sub-rotinas já prontas na ROM do seu computador, tendo cada uma seu papel definido. Essas sub-rotinas podem ser requisitadas a qualquer momento através do uso da instrução JSR e como estão contidas na ROM, jamais poderão ser apagadas ou ter seus valores alterados (tabelas VII e VIII).

TABELA VII

Endereços	Funções
F800	Exibe na tela caractere Plot
F819	Exibe na tela uma linha horizontal
F828	Exibe na tela uma linha vertical
FB40	Estabelece o modo gráfico
FC58	Limpa a tela
FD1B	Gera um número aleatório
FD35	Espera uma tela ser pressionada
FD8E	Gera RETURN
FDDA	Mostra o conteúdo do acumulador (HEX)
FF3A	Gera um sinal sonoro

As sub-rotinas da ROM da tabela VII, estão armazenadas na memória ROM do TK 2000 e podem ser explicadas com mais detalhes com o auxílio do seu manual técnico.

TABELA VIII

TK 2000

Endereços	Funções
F043	Verifica o teclado
F800	Exibe na tela um plot em baixa resolução
F819	Exibe na tela uma linha horizontal
F828	Exibe na tela uma linha vertical
F832	Limpa a tela inteira
F836	Limpa a tela com exceção a área de texto
F869	Exibe na tela o número da cor de um bloco
F85F	Incrementa o número da cor
F864	Determina a cor do bloco
F941	Mostra os conteúdos de A e X
F948	Coloca na tela três espaços em branco
F94A	Coloca na tela de 1 a 256 espaços
FBDD	Gera um Sinal Sonoro (BEEL)
FCA8	Atraso (WAIT)
FD67	Obtém uma linha de entrada
FD6A	Obtém uma linha com o símbolo " "
FD6F	Obtém um linha sem o símbolo " "
FD8B	Limpa o fim e gera RETURN
FD8E	Gera um RETURN
FDDS	Mostra o conteúdo do acumulador (HEX)
FDE3	Mostra a metade menos significativa do conteúdo do acumulador
FDED	Mostra o caractere cujo código está no acumulador
FE80	Muda o vídeo para modo inverso
FE84	Muda o vídeo para modo normal
FFLD	Mostra a palavra "ERR"
FF3A	Gera um sinal sonoro (BEEP)
FF3F	Carrega os registradores
FF4A	Salva os registradores

Vamos agora apresentar alguns programas curtos e simples, para que você possa entender o funcionamento de cada um deles, através da análise instrução por instrução.

Em todos eles é usada uma mesma sub-rotina da ROM, que tem como finalidade imprimir o valor do acumulador no final do programa.

Antes de rodá-lo, porém, tente descobrir o valor do acumulador de cada um deles, analisando os programas um a um.

Figura 2

STA ($ 10), Y

Supondo conteúdo de { Y → 07; $ 0010 → 30; $ 0011 → 10 }

$ 10 + 1 → $ 11
$ 0080 → # 30
$ 0011 → # 10
} conteúdo dos endereços

Endereço formado: $ 1030 + Y (= 07) $ 1037

STX — Armazena o valor do indexador X na memória (tabela II)

TABELA II

Código usado	Formato	Bytes usados
86	STX OPER	2
96	STX OPER, Y	2
8E	STX OPER	3

As instruções STX funcionam de modo idêntico às instruções STA, com uma única diferença: a de haver troca de registros, ou seja, a STX utiliza o indexador X e a STA utiliza o registro acumulador.

As instruções STX podem apresentar três formatos diferentes:

STX OPER — Endereçamento direto (página zero). Funcionamento idêntico à instrução STA OPER. Por exemplo: STX $ 30 — Coloca o valor do registro X no endereço $ 30.

STX OPER, Y — Endereçamento indexado por Y (página zero). Por exemplo: STX $ 50 Y — Supondo Y = 60, o operando da instrução é adicionado ao valor do registro Y. ($ 50 + '60' = $ B0), ou seja, o valor do registro X é colocado no endereço $ B0.

STX OPER — Endereçamento direto (Absoluto). Funcionamento idêntico à instrução STA OPER (Absoluto), levando em consideração a troca de registros A por X. Por exemplo:

STX $ 900 — Supondo X = 15, coloca o valor de X, ou seja, o número # 15 no endereço $ 9000.

STY — Armazena o valor do indexador Y na memória (tabela III).

TABELA III

Código usado	Formato	Bytes usados
84	STY OPER	2
94	STY OPER, X	2
8C	STY OPER	3

Todas as instruções que compõem a STY são de funcionamento idêntico às instruções STX, com a única diferença de haver a troca de registradores X por Y.

JSR — Salto para uma sub-rotina.

TABELA IV

Código usado	Formato	Bytes usados
20	JSR OPER	3

A instrução JSR significa salto para uma sub-rotina e pode ser comparada à instrução GOSUB em BASIC.

As duas podem ser comparadas porque tanto uma como a outra saltam a uma sub-rotina.

As sub-rotinas podem ser classificadas como um "miniprograma" que não faz parte do programa principal e são chamadas a qualquer momento através da instrução JSR.

Uma sub-rotina pode ser requisitada uma, duas ou mais vezes pelo programa principal.

Quando um programa que está em plena execução encontra uma instrução do tipo JSR é como se ele parasse de executá-lo e passasse a executar outro programa.

Se pensássemos um pouco, acharíamos que faltava alguma instrução para que no final da execução de uma sub-rotina houvesse o retorno ao programa principal. Na verdade, existe essa instrução de retorno, que é simbolizada pela sigla RTS, que significa retorno de uma sub-rotina. Em outras palavras, esta instrução finaliza a execução de uma sub-rotina e retorna ao programa principal.

RTS — Retorno de uma sub-rotina (tabela V).

TABELA V

Código usado	Formato	Bytes usados
60	RTS	1

Figura 3

```
0800 — 91 05      LDA = $ 05     }
0802 — 8S 50      STA $ 50       } programa principal
0804 — 20 20 08   JSR $ 0820     }
0807 — A6 69      LDX $ 69       }

0820 — A2 30      LDX # $ 30     }
0822 — 86 51      STX $ 51       } sub-rotina
0824 — 60         RTS            }
0825 — BC 21 10   LDY $ 1021, X  }
```

Vamos, através do exemplo da figura 3, analizar o funcionamento das instruções JSR e RTS. Supondo que o programa tenha início no endereço $ 0800:

— O acumulador é carregado com o valor '05'.
— O conteúdo do acumulador, ou seja '05', é armazenado no endereço $ 50.
— O programa passa agora a ser executado a partir do endereço $ 0820.
— O registro X é carregado com o valor '30'.
— O conteúdo do registro X, ou seja, '30' é armazenado no endereço $ 51.
— Encontra a instrução de retorno, voltando imediatamente após a instrução JSR, ou seja, ao endereço $ 0807.
— O registro X é carregado com o conteúdo do endereço $ 69.
— Segue o programa.

JMP — Salta para um novo endereço (tabela VI).

Programa 1

```
0800 — A9 05         LDA # $ 05
0802 — 20 DA FD      JSR $ FDDA
0805 — 60            RTS
```

Programa 2

```
0800 — A2 07         LDX # $ 07
0802 — A9 50         LDA # $ 50
0804 — 95 70         STA $ 70, X
0806 — A5 77         LDA $ 77
0808 — 20 DA FD      JSR $ FDDA
080B — 60            RTS
```

Programa 3

```
0800 — A9 00         LDA # $ 00
0802 — 85 6C         STA $ 6C
0804 — A9 03         LDA # $ 03
0806 — 8S 6D         STA $ 6D
0808 — A2 1C         LDX # S 1C
080A — A0 10         LDY # S 10
080C — 8C 00 03      STY $ 0300
080F — A1 50         LDA ($ 50, X)
0811 — 20 DA FD      JSR $ FDDA
0814 — 60            RTS
```

Programa 4

```
0800 — A9 10         LDA # $ 10
0802 — 85 51         STA $ 51
0804 — A9 30         LDA # $ 30
0806 — 8S 50         STA $ 50
0808 — A2 00         LDX # $ 00
080A — 86 01         STX $ 01
080C — A0 10         LDY # $ 10
080E — 84 02         STY $ 02
0810 — A9 10         LDA # $ 10
0812 — 85 30         STA $ 30
0814 — A0 0S         LDY # $ 05
0816 — B6 2B         LDX $ 2B, Y
0818 — 81 40         STA ($ 40, X)
081A — A4 50         LDY $ 50
081C — B1 01         LDA ($ 01), Y
081E — 20 DA FD      JSR $ FDDA
0821 — 60            RTS
```

Para entrar no modo monitor e digitar os programas, digite LM, seguido por RETURN. Digite os endereços e os códigos hexadecimais, seguidos por RETURN após cada linha. Não digite os mneumônicos, eles são apenas um auxiliar. Para verificar as listagens digite endereço inicial, seguido por um ponto e o endereço final seguido por L, numa única linha.

TENTE ESTA

Estes são para o TK 83/85

Daniel Nordemann

Estes dois programas curtíssimos são para enfeitar a tela da sua televisão com quadros animados. Operar em Slow.

```
  1 PRINT AT 21*RND,31*RND;CHR$
(6*RND);
  2 RUN
```

e:

```
 1 PLOT 63*RND*RND,43*RND*RND
 2 UNPLOT 63*RND,43*RND
 3 RUN
```

São fáceis de entender e podem ser modificados à vontade (diminuir os quadros, usar menos caracteres, até um só se quiser, misturar os dois programas, alterar a densidade de caracteres.

SPACE EGGS

Um jogo emocionante para TK 2000

Que tal exterminar monstros incubados em ovos no espaço? Este é basicamente o enredo deste interessante jogo fabricado pela Cibertron.

Marcos Lorenzi

Na seção "Analisando" deste mês, estamos apresentando o jogo "Space Eggs", para o TK 2000, fabricado pela Cibertron.

O jogo, desenvolvido totalmente em linguagem de máquina, foi nos enviado para análise em sua primeira versão. Já está no mercado uma segunda versão basicamente igual à primeira, tendo como diferença a forma de carregamento, que se tornou mais simples.

A embalagem que acompanha a fita é bem simples, onde consta o nome da empresa, do produto e a que computadores o programa é compatível.

Acompanhando a fita vem o manual de instruções, que está bem detalhado facilitando a sua compreensão, não deixando transparecer dúvidas com relação ao processo de jogo.

O programa está gravado em ambos os lados apenas por motivo de segurança, caso uma das versões apresente algum problema. Notamos que a qualidade da fita é muito boa.

Na própria fita está escrito o código de carregamento totalmente em linguagem de máquina. Usa-se o comando LM mais o código anexo, (já a nova versão utiliza o comando LOAD, como auto-start).

No jogo, você encontra-se no espaço e terá que enfrentar criaturas que estão incubadas em ovos durante sua jornada.

Seu objetivo é destruí-las antes que estas acabem fazendo o mesmo com você. Para isso, você dispõe de três naves, ou seja, terá três chances para destruir o inimigo.

O jogo consiste em quatro fases e para se conseguir passar de um estágio ao outro é preciso vencer a todas criaturas, sem que sua nave seja abatida.

O quadro abaixo mostra os valores respectivos de cada fase:

FASES	PONTOS
1º	15
2º	30
3º	45
4º	80

Em cada fase nota-se um grau de dificuldade crescente para se conseguir vencer o inimigo.

Observamos que nos três primeiros estágios, os graus de dificuldade são parecidos, apenas havendo diferença na intensidade, velocidade e o modo de como são elaborados os ataques.

Na última fase notamos ser a mais difícil, onde o inimigo busca destruí-lo de uma forma veloz e objetiva.

Acoplando duas naves

Passando pelas quatro fases e conseguindo o feito, o próximo estágio será misto, ou seja, você confrontará com os quatro tipos de criaturas juntos.

Outra meta que você terá como desafio é tentar acoplar outra nave à sua, onde seu poder de fogo poderá chegar a ser de até cinco.

Para que o jogador consiga acoplar outra nave, ele deve alcançar os mil pontos e ter uma nave extra ao mesmo tempo. A cada mil pontos obtidos o participante tem a chance de acoplar outra nave além daquela que já possui, sendo três o máximo de espaçonaves que podem ser acopladas e o poderio de fogo, caso você consiga, pode chegar a ser de cinco tiros, sendo o limite máximo.

Você conseguindo superar todas essas barreiras, pode-se dizer que sua missão está cumprida, restando apenas manter seu nível de ataque para ter maior facilidade em derrotar seu inimigo.

O Space Eggs contém dois placares, que estão localizados no topo da tela de jogo e são de fácil leitura.

O primeiro placar mostra os pontos obtidos pelo jogador até o momento em que o mesmo estiver em jogo.

O segundo registra o maior número de pontos conseguidos pelo jogador (RECORD). Este será alterado se o mesmo conseguir ultrapassar a sua maior marca ficando registrada a nova.

Os números são de fácil leitura e de tamanho adequado, proporcionando ao jogador uma clara identificação. Apenas uma observação: o número oito é facilmente confundido com o nove, causando certa dificuldade ao lê-lo.

A tela de apresentação do Space Eggs está muito bem elaborada, em letras grandes e legíveis, contendo ainda a característica das criaturas de cada fase com seus respectivos valores.

A segunda tela, ou demonstrativo, deveria ser apresentada de uma forma mais completa, onde apenas aparece a nave acoplada a outra, movendo-se horizontalmente sem dar disparos e as criaturas incubadas em ovos em movimentos circulares.

Esta tela de demonstrativo, apenas como exemplo para se ter uma noção de como é o jogo, poderia apresentar alguns detalhes mais específicos.

A tela de jogo está representada em dois planos, ou seja, nota-se perfeitamente que os movimentos da nave em relação ao fundo, que é o espaço, são contrários, dando um aspecto mais realista ao jogo.

Ao fazermos uma comparação do som em relação ao estilo do jogo, notamos que não há uma combinação perfeita de efeitos, tirando o que seria a característica principal de um "game".

Uma melhor elaboração e distribuição de graves e agudos nas diferentes fases do jogo, conseqüentemente o tornaria mais apresentável, (por sinal seu nível é muito bom).

Conclusão

Por ser bem elaborado, o Space Eggs consegue prender a atenção fazendo com que você fique horas diante da TV tentando superar seu inimigo.

Um jogo que além de agradar aos mais jovens, também atingirá os adultos, sendo estes, às vezes, mais interessados em videogames do que os jovens.

Programas TK 2000

Renumerador de linhas

Wladimir Bastos

Eis um pequeno truque que permite que você renumere um programa em BASIC, sem corrigir os endereços de GOTO e GOSUB. Apesar disso, este é um excelente ponto de partida para você começar a desenvolver aplicativos.

Este programa reorganiza os números que rotulam as linhas de um programa em BASIC no TK 2000. Ele permite que se altere o número da primeira linha e se dê um incremento constante, permitindo que se aumente o espaço entre linhas para correção ou para dar-se um acabamento mais estético.

Após ter digitado o programa que deverá ser renumerado, devemos digitar o programa renumerador, a partir de uma linha maior que a última linha do programa anterior.

Então devemos escolher qual o valor da primeira linha e o incremento a ser dado (estes valores serão pedidos pelo programa renumerador) e digitar GOTO xx, sendo xx o número da primeira linha do programa renumerador, seguido pela tecla RETURN. Espere alguns segundos e o programa estará renumerado. *Atenção para os desvios* (200 GOTO 500, 200 IF A = Ø THEN GOTO 500, etc.). *Eles permanecerão com seus valores originais,* sem serem afetados pelo programa renumerador. Portanto, logo após a renumeração, estes valores devem ser trocados pelos novos. 'ores.

Programa Renumerador

```
100  INPUT "QUAL O VALOR DA PRIMEIRA L
INHA ?";A
200  INPUT "QUAL O VALOR DO INCREMENTO
 ?";B
300 X =  -  FRE (0) - 24578
400  FOR I = 2048 TO X: IF  PEEK (I) =
0 THEN  IF A > 255 THEN  POKE I +
3,A -  INT (A / 256) * 256: POKE
I + 4, INT (A / 256):A = A + B:I = I + 5: NEXT
500  IF  PEEK (I) = 0 THEN  POKE I + 3
,A: POKE I + 4,0:A = A + B:I = I + 5
600  NEXT
```

Programa a ser renumerado

```
10  REM  --- PROGRAMA PARA ACHAR QUAND
O H <> 0 ---
20 H =  INT (( RND (1) * 3) / 2)
30  IF H = 0 THEN 20
40  PRINT " O VALOR E MAIOR QUE ZERO "
;H
50  END
```

Programa original mais renumerador

```
10  REM  --- PROGRAMA PARA ACHAR QUAND
O H <> 0 ---
20 H =  INT (( RND (1) * 3) / 2)
30  IF H = 0 THEN 20
40  PRINT " O VALOR E MAIOR QUE ZERO "
;H
50  END
60  INPUT "QUAL O VALOR DA PRIMEIRA LI
NHA ?";A
70  INPUT "QUAL O VALOR DO INCREMENTO
?";B
80 X =  -  FRE (0) - 24578
90  FOR I = 2048 TO X: IF  PEEK (I) =
0 THEN  IF A > 255 THEN  POKE I +
3,A -  INT (A / 256) * 256: POKE
I + 4, INT (A / 256):A = A + B:I = I + 5: NEXT
100  IF  PEEK (I) = 0 THEN  POKE I + 3
,A: POKE I + 4,0:A = A + B:I = I + 5
110  NEXT
```

Após ter digitado GOTO 60 e RETURN obteremos:

```
0  REM  --- PROGRAMA PARA ACHAR QUANDO
 H <> 0 ---
5 H =  INT (( RND (1) * 3) / 2)
10  IF H = 0 THEN 20
15  PRINT " O VALOR E MAIOR QUE ZERO "
;H
20  END
25  INPUT "QUAL O VALOR DA PRIMEIRA LI
NHA ?";A
30  INPUT "QUAL O VALOR DO INCREMENTO
?";B
35 X =  -  FRE (0) - 24578
40  FOR I = 2048 TO X: IF  PEEK (I) =
0 THEN  IF A > 255 THEN  POKE I +
3,A -  INT (A / 256) * 256: POKE
I + 4, INT (A / 256):A = A + B:I = I + 5: NEXT
45  IF  PEEK (I) = 0 THEN  POKE I + 3,
A: POKE I + 4,0:A = A + B:I = I +
5
50  NEXT
```

A seguir deveremos trocar a linha 1Ø por 1Ø IF H = Ø THEN 5 e apagar as linhas 25 a 50.

Desafie um amigo para ver quem, controlando um grupo de balões carregados de explosivos, derruba o maior número possível de pedras numa montanha.

Dive Bomber

Adaptado por Mário Folli

Dive bomber é um programa conhecido por muitos possuidores do Apple. Pode ser um jogo interessante, que faz uso de muitas instruções gráficas e sonoras, achamos que ele deveria ser adaptado para o TK 2000.

Neste jogo, seu objetivo é tentar superar seu adversário na destruição de rochas que se encontram numa montanha, fazendo com que um balão carregado de explosivos se choque com ela.

O programa chama alternativamente os dois jogadores para controlarem seus respectivos balões.

Comentários sobre as linhas de programa

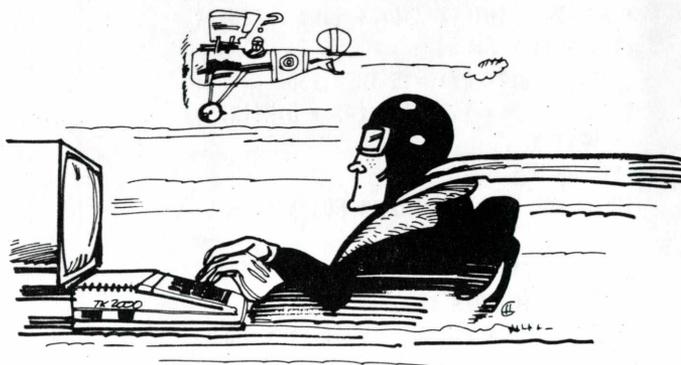
4 a 210 — Apresentação do jogo e instruções.
220 a 400 — Carrega a tabela de figuras na memória, a partir do endereço 16384 (decimal).
440 a 630 — Desenha a tela do jogo.
640 a 750 — Mostra os nomes dos jogadores, placar e erros de alvo.
760 a 810 — Efetua a contagem de tempo para o jogador se preparar.
820 a 930 — Movimenta grupo de balões pela tela.
950 a 1060 — Movimenta um dos balões para baixo, sem verificar se atingiu as rochas.
1070 a 1150 — Movimenta o balão do restante do caminho, verificando se atingiu as rochas.
1160 a 1170 — Verifica se o balão que caiu atingiu a montanha, sem destruir as rochas (situação de erro).
1190 a 1280 — Verifica quantas rochas foram atingidas, incrementando o placar do jogador.
1290 a 1390 — Explode as rochas.
1400 a 1550 — Faz as rochas não atingidas e que perderam a sustentação cairem.
1560 — Verifica se todas as rochas foram destruídas.
1610 a 1850 — Finaliza a partida, congratulando o vencedor.

O programa

```
3  REM  DIVE BOMBER
4  HOME : FOR I = 1 TO 22
5  VTAB I: PRINT "********  D I V E  B
 O M B E R  ********"
10  SOUND I,40
15  VTAB I: PRINT "
                    "
20  NEXT I
30  DIM HX(9),HY(9)
40  DIM B(45,10): MP : HOME
50  VTAB 1: HTAB 6: PRINT "DIVE BOMBER
 - (INSTRUCOES)"
60  PRINT  TAB( 6)"===========   =====
======"
70  VTAB 4: PRINT "  - O SEU OBJETIVO
E DE SEU OPONENTE E'"
80  VTAB 6: PRINT "DESTRUIR O MAIOR NU
MERO POSSIVEL DE"
90  VTAB 8: PRINT "ROCHAS (CIRCULOS BR
ANCOS)."
100  VTAB 10: PRINT " - UMA FAIXA BRAN
CA SOBRE O NOME DO"
110  VTAB 12: PRINT "JOGADOR,INDICA A
SUA VEZ DE JOGAR."
120  VTAB 14: PRINT "  -PARA LARGAR A
BOMBA,PRESSIONE A"
130  VTAB 16: PRINT "BARRA DE ESPACOS.
"
140  VTAB 18: PRINT "  - CADA JOGADOR
PODE ERRAR NO MAXIMO"
150  VTAB 20: PRINT "5 VEZES (QUANDO F
ICARA SEM BOMBAS)."
160  VTAB 22: HTAB 6: PRINT "(PRESSION
E QUALQUER TECLA)"
170  GET A$: HOME
180  VTAB 6: INPUT "QUAL O NOME DO JOG
ADOR 1? ((10)";J1$
190  IF  LEN (J1$) > 9 THEN 180
200  VTAB 8: INPUT "QUAL O NOME DO JOG
ADOR 2? ((10)";J2$
210  IF  LEN (J2$) > 9 THEN 200
220  VTAB 11: HTAB 9: PRINT "* AGUARDE
 UM MOMENTO *"
230 P = 16384
240  POKE 232,0: POKE 233,64
250  READ I$: IF I$ = "END" THEN 410
260  FOR A = 1 TO  LEN (I$) - 1 STEP 2

270 P$ =  MID$ (I$,A,2)
280 H =  ASC ( LEFT$ (P$,1))
290 L =  ASC ( RIGHT$ (P$,1))
300 H = H - 48: IF H > 9 THEN H = H -
7
310 L = L - 48: IF L > 9 THEN L = L -
```

```
7
320  POKE P,H * 16 + L
330 P = P + 1
340  NEXT A
350  GOTO 250
360  DATA  "030008001F003700"
370  DATA  "252D2C2D2E35353F3F3F3F3735
35353525252525252D2C2D2E35353F3F3
F3F37353535352525252525250500"
380  DATA  "25253F3F24242C2D253C3E2627
372E2D3536363F3F2E2E00"
390  DATA  "1225253C3C37372E2E0500"
400  DATA  "END"
410 S1 = 0:S2 = 0:M1 = 0:M2 = 0
420  VTAB 14: HTAB 6: PRINT "< PRESSIO
NE QUALQUER TECLA >"
430  GET A$: POKE  - 16368,0
440  HOME
450  HGR
460  ROT= 0: SCALE= 1
470 HT = 0
480  FOR X = 1 TO 40
490  READ R: IF R = 0 THEN 560
500  HCOLOR  = 1
510  FOR DX = (X - 1) * 6 TO (X - 1) *
6 + 6
520  HPLOT DX,159 TO DX,159 - R * 6
530  NEXT DX
540  FOR Y = 1 TO R:B(X,Y) = 5: NEXT Y

550  HCOLOR  = 3
560  FOR Y = R + 1 TO 10
570  DRAW 3 AT (X - 1) * 6 + 3,159 - (
Y - 1) * 6 - 3
580 B(X,Y) = 1
590  NEXT Y
600 HT = HT + 10 - R
610  NEXT X
620  DATA  9,8,7,5,4,3,2,0,0,1,2,3,4,3
,3,4,5,6,5,4
630  DATA  4,5,6,5,4,3,3,4,3,2,1,0,0,2
,3,4,5,7,8,9
640  HOME
650  VTAB 22: PRINT  TAB( 33);: INVERSE
: PRINT "DIVE";: NORMAL : PRINT " "
660  VTAB 23: PRINT  TAB( 32);: INVERSE
: PRINT "BOMBER";: NORMAL : PRINT " "
670 SW =  - 1
680  IF SW = 1 THEN SW =  - 1:F = 230:
T = 10: GOTO 700
690 SW = 1:F = 10:T = 230
700  IF M1 = 5 AND SW = 1 THEN 680
710  IF M2 = 5 AND SW =  - 1 THEN 680
720  VTAB 22: IF SW = 1 THEN  INVERSE
730  PRINT J1$;: NORMAL : PRINT  TAB(
11);"PLACAR:";S1; TAB( 22);"ERROS
:";M1
740  VTAB 23: IF SW =  - 1 THEN  INVERSE

750  PRINT J2$;: NORMAL : PRINT  TAB(
11);"PLACAR:";S2; TAB( 22);"ERROS
:";M2
760  FOR CR = 3 TO 1 STEP  - 1
770  SOUND 40,20
780  VTAB 21: HTAB 14: PRINT "TEMPO:":
 INVERSE : VTAB 21: HTAB 20: PRINT "0":
VTAB 21: HTAB 21: PRINT CR: NORMAL
790  POKE  - 16368,0
800  FOR P = 1 TO 300: NEXT P
810  NEXT CR
820  VTAB 21: PRINT  TAB( 14)"
 "
830  FOR IN = F TO T STEP SW * 3
840  SOUND Y,5
850  HCOLOR  = 3: DRAW 1 AT IN - 4,10
860  DRAW 2 AT IN,25
870 AA =  PEEK (39)
880  IF AA = 48 THEN 900
890  POKE  - 16368,0: GOTO 950
900  HCOLOR  = 0
910  DRAW 1 AT IN - 4,10
920  DRAW 2 AT IN,25
930  NEXT IN
940  GOTO 1580
950  HCOLOR  = 0: DRAW 1 AT IN - 4,10
960  DRAW 2 AT IN,25
970 X = IN
980  FOR Y = 25 TO 99 STEP 2
990  SOUND Y,1
1000  HCOLOR  = 0: DRAW 2 AT X,Y - 2
1010 X = X + SW * 1.75
1020  IF X <  = 0 THEN X = 240: GOTO 1
040
1030  IF X >  = 240 THEN X = 0
1040  HCOLOR  = 3: DRAW 2 AT X,Y
1050  NEXT Y
1060  HCOLOR  = 0: DRAW 2 AT X,99
1070  FOR Y = 99 TO 158 STEP 2
1080  SOUND Y,1
1090  HCOLOR  = 0: DRAW 2 AT X,Y - 2
1100  IF B( INT (X / 6) + 1, INT ((159
 - Y) / 6)) > 0 THEN 1160
1110 X = X + SW * 1.75: IF X < 0 THEN
X = X * - 1
1120  HCOLOR  = 3: DRAW 2 AT X,Y
1130  NEXT Y
1140  HCOLOR  = 0: DRAW 2 AT X,157
```

```
1150  GOTO 1580
1160 RX =  INT (X / 6) + 1:RY =  INT (
(159 - Y) / 6)
1170  IF B(RX,RY) = 5 THEN 1580
1180  FOR X = 1 TO 9:HX(X) = 0:HY(X) =
0: NEXT X
1190 HP = 1
1200  FOR Y =  - 1 TO 1: FOR X =  - 1 TO
 1
1210  IF RX + X < 1 OR RX + X > 40 OR
RY + Y > 10 THEN 1280
1220  IF B(RX + X,RY + Y) <  > 1 THEN
1280
1230 HX(HP) = RX + X:HY(HP) = RY + Y
1240 B(RX + X,RY + Y) = 0
1250  IF SW = 1 THEN S1 = S1 + 1: GOTO
1270
1260 S2 = S2 + 1
1270 HP = HP + 1
1280  NEXT X: NEXT Y
1290 HP = HP - 1
1300  FOR FL = 1 TO 10
1310  HCOLOR  = 3
1320  FOR FO = 1 TO HP
1330  DRAW 3 AT HX(FO) * 6 - 3,162 - H
Y(FO) * 6
1340  NEXT FO
1350  HCOLOR  = 0
1360  FOR FO = 1 TO HP
1370  DRAW 3 AT HX(FO) * 6 - 3,162 - H
Y(FO) * 6
1375  SOUND 5,HP
1380  NEXT FO
1390  NEXT FL
1400 FS = RX - 1:TS = RX + 1
1410  IF FS < 1 THEN FS = 1
1420  IF TS > 40 THEN TS = 40
1430  FOR S = FS TO TS
1440  FOR Y = 2 TO 10
1450  IF B(S,Y) <  > 1 THEN  GOTO 1540

1460  FOR C = Y - 1 TO 1 STEP  - 1
1470  IF B(S,C) <  > 0 THEN 1500
1480  NEXT C
1490 C = 0
1500  IF C + 1 = Y THEN 1540
1510 B(S,C + 1) = B(S,Y):B(S,Y) = 0
1520  HCOLOR  = 0: DRAW 3 AT S * 6 - 3
,162 - Y * 6
1530  HCOLOR  = 3: DRAW 3 AT S * 6 - 3
,162 - (C + 1) * 6
1540  NEXT Y
1550  NEXT S
1560  IF (S1 + S2) / HT =  INT ((S1 +
S2) / HT) THEN 1610
1570  GOTO 680
1580  IF SW = 1 THEN M1 = M1 + 1
1590  IF SW =  - 1 THEN M2 = M2 + 1
1600  IF M1 + M2 < 10 THEN 680
1605  IF (S1 + S2) / HT =  INT ((S1 +
S2) / HT) THEN 91900
1610  HCOLOR  = 1
1620 A = 0:B = 0
1630 C = 240:D = 104
1640  HPLOT A,B TO C,B
1650  HPLOT A,D TO C,D
1660  HPLOT X,D
1670  SOUND 80,50
1680 C = C - 8:D = D - 8
1690 A = A + 8:B = B + 8
1700  IF D = 64 THEN 1720
1710  GOTO 1640
1720  IF S1 > S2 THEN G$ = J1$
1730  IF S1 < S2 THEN G$ = J2$
1740  IF S1 = S2 THEN G$ = "AOS DOIS"
1750 A =  LEN (G$)
1760 H =  INT ((28 - A) / 2)
1770  VTAB 7: HTAB H: INVERSE : PRINT
 TAB( H)"PARABENS "; TAB( H + A)G
$: NORMAL
1780  SOUND 100,100 TO 100,100 TO 125,
100 TO 150,60
1790  FOR I = 1 TO 1000: NEXT I
1800  VTAB 6: PRINT  TAB( 16)"FINAL"
1810  VTAB 7: PRINT  TAB( 17)"DA
  "
1820  VTAB 8: PRINT  TAB( 15)"PARTIDA"

1830  SOUND 100,100 TO 100,100 TO 150,
100 TO 100,60
1840  FOR I = 1 TO 700: NEXT I
1850  TEXT
```

Duas dicas para o Apple e o TK 2000

Ocultando uma mensagem
Efeitos gráficos

No artigo anterior falei alguma coisa sobre como esconder sua listagem dos olhos indiscretos. Aqui eu vou apresentar uma outra maneira bastante interessante. Vá ao seu computador Apple II ou compatível e tecle as seguintes linhas:

```
 ] NEW
 ] CALL-151
* 800:00 18 08 0A 00 BA 22 CC
      C5 C9 C1 A0 CD C9 C3 D2
      CF C8 CF C2 C2 D9 22 00
      00 00
* C (ou Reset)
```

Liste agora o seu programa. Esquisito não? Dê o RUN para ver o que acontece. Muito bem, conseguimos escrever uma linha de PRINT cujo conteúdo é ilegível. A sua utilidade é óbvia, mas como utilizá-lo para um programa normal? Façamos o seguinte:

```
 ] 10 PRINT "LEIA MICROHOBBY"
 ] CALL-151
* 800.819
800-00 18 08 0A 00 BA 22 4C
808 - 45 49 41 20 4D 49 43 52
810 - 4F 48 4F 42 42 59 22 00
818 - 00 00
```

Compare esta listagem com os dados que você entrou da primeira vez. A diferença está no bit mais significativo, a partir do primeiro "22" da listagem. Na verdade, a primeira listagem é idêntica à segunda com exceção destes bits mencionados. O que fizemos foi modificar o bit mais significativo dentro da mensagem entre as duas ASPAS (código 22). Então, para reproduzir o efeito que já mostramos, basta entrar no monitor, modificar o bit mais significativo da mensagem entre as aspas e pronto. A sua mensagem está escondida.

Explicado o funcionamento deste pequeno truque, você pode alterar pelo monitor a sua mensagem, bastando localizar o BA 22 (PRINT) e trocar o bit mais significativo. Para quem quiser automatizar essa troca, basta escrever uma rotina que faça isto automaticamente. Deixo isto como exercício para os leitores.

Efeitos gráficos

Se você tem uma TV em cores, experimente o seguinte programa:

```
10  HGR2
20 T = 2
30 OF = 140
40 EI = 95
50 N = 80
60 OX = 140
70 OY = 94
80  HCOLOR  = 4
90 M = 130
100  FOR K = 1 TO 100
110 X =  INT ( RND (1) * M + OF)
120 Y =  INT ( RND (1) * N + EI)
130  FOR I = 1 TO N STEP T
140 X = X - T:Y = Y - T
150 RY = T * EI - Y
160 RX = T * OF - X
170  HPLOT OX,OY TO X,Y
180  HPLOT XO,OY TO RX,Y
190  HPLOT XO,OY TO RX,RY
200  HPLOT XO,YO TO X,RY
210  NEXT I
220  HCOLOR  = 7 *  RND (1)
230 OX = X:OY = Y:YO = RY:XO = RX
240  NEXT K
250  GOTO 10
```

Ko Ming Cho

Ko Ming Cho é engenheiro Eletrônico formado pelo Ita e diretor da Micro Mania.

Devido sua compatibilidade com modelos anteriores da Texas (TI 58 e TI 59), esta calculadora permite ao seu usuário um grande número de aplicações nos mais diversos campos.

A TI-66, uma calculadora programável versátil

Fabio Augusto Polônio

Caracterizada por sua extrema facilidade de programação, a calculadora TI-66 TEXAS, fabricada pela Texas Instrumentos Eletrônicos do Brasil, inclui em seu sistema operacional diversas funções algébricas, logarítmicas e trigonométricas, além de funções específicas para cálculos estatísticos.

Possui 47 teclas, na sua maioria com dupla função. A primeira função está representada na tecla e a segunda está no corpo da calculadora logo acima da tecla.

Para que seja acessada a segunda função, devemos antes usar a tecla 2nd (second) mais a tecla desejada.

Sua memória é particionada em memória de programa (passos) e memória de dados.

O usuário pode optar pela reserva de áreas de programa ou de dados, através da tecla (2nd). Part. Quando a calculadora está desligada, o conteúdo da memória é preservado através da característica operacional de memória constante. Pode-se reservar até 64 endereços de memória de dados ou 512 passos de programa. Para cada endereço de memória reservado decrementa-se em 8 passos a área de programa. Assim, para 32 endereços de memória temos 255 passos de programa; para 33 endereços 247 passos e assim sucessivamente até 64 memórias e nenhum passo, ou 512 passos e nenhuma memória de dados.

Funções fundamentais de Controle de programas

Para introduzir o modo de programação aciona-se a tecla LRN (learn-informar). Pressionando-se uma só vez esta tecla coloca a calculadora na modalidade de informação.

Elimina-se o modo de programação sempre que pressionadas as teclas 2nd CP.

A tecla R/S reverte o estado de processamento. Pressionando-se R/S dá-se início ao processamento a partir de onde estiver o apontador do programa. Para estabelecer o apontador no início aciona-se RST (restabelecer), que elimina o conteúdo do registrador de retorno também.

É possível a interrupção do programa através do comando PAUSE, que interrompe o processamento por pouco mais de um segundo e faz com que o valor contido no registrador do visor seja apresentado no momento. Manter esta tecla pressionada durante a execução de um programa faz com que cada passo do programa seja apresentado.

Há cinco teclas de letras que representam dez labels. Um label serve somente como um ponto de identificação em um programa. A execução poderá procurar um label e ir até ele, mas não haverá nenhum significado numérico (equivale ao número de linha de um programa em BASIC).

Ela aceita dois tipos de notação científica (múltiplos de potências de 10). A tecla EE aciona o modo de notação científica e 2nd ENG aciona a notação de engenharia (as potências são múltiplas de 3) que é usada mais comumente em cálculos que exigem manipulação de unidades, (por exemplo, 10^{-6} = micro, 10^{-9} = nano, 10^{-12} = pico).

Além das funções algébricas e transcendentais, a TEXAS TI-66 possui funções estatísticas pré-definidas. A calculadora utiliza as memórias de 1 a 6 e o registrador *t* para efetuar cálculos estatísticos.

2nd CSR (clear Statistics Register) inicializa a calculadora para cálculos estatísticos, zerando as memórias e o registrador *t*.

2nd $\Sigma +$, acumula cada par de dados e, usando a tecla INV, elimina qualquer ponto de dados indesejável. Pode-se calcular diretamente a variância, desvio-padrão e média aritmética.

Conclusão

Com um bom conjunto de funções pré-definidas e permissividade de programação, esta calculadora permite bom desempenho e rapidez de operação, sem a necessidade do usuário ser um exímio programador. Com poucos conhecimentos pode-se explorar bem todos os seus recursos.

Análise Nodal

Wilson José Tucci
José Eduardo Moreira

Em poucos ramos de engenharia, a matemática é tão útil como na análise de circuitos elétricos. Teorias muito bem desenvolvidas simplificam, em muito, esta tarefa.

O problema consiste basicamente em reconhecer os componentes do circuito, montar, de forma semi-automática, um sistema de equações e resolvê-lo.

Faremos uma exposição da análise de circuitos resistivos lineares. Neste caso, recaímos na resolução de um sistema de equações lineares, o que é muito simples de ser feito com uma calculadora avançada como a HP-41.

Assumiremos que o leitor tenha uma familiaridade mínima com circuitos elétricos.

Um circuito elétrico é constituido de nós e ramos. Um nó deve ser escolhido como referência. Esta escolha é, em princípio, arbitrária, mas é conveniente escolher o "terra" como nó de referência. Depois disso todos os nós são numerados ("terra" é Ø). A tensão de cada nó, em relação ao nó de referência, é chamada de tensão nodal, e ela é determinada.

Num primeiro exemplo chegaremos lentamente ao sistema de equações na sua forma matricial mais prática.

Na análise nodal não é comum usar-se os valores das resistências R, mas seu inverso $G = 1/R$, chamado de condutância. Como $V = RI$, temos $I = GV$. Essa relação será usada mais adiante.

A análise nodal parte da aplicação da primeira lei de Kirchoff aos nós que não são de referência do circuito. A primeira lei de Kirchoff diz que a somatória das correntes que chegam e saem em um nó é zero (isso quer dizer que em um nó não há acúmulo ou drenagem de carga).

Consideremos o i-ésimo nó não de referência de uma rede, contendo apenas resistores e geradores independentes (figura 1).

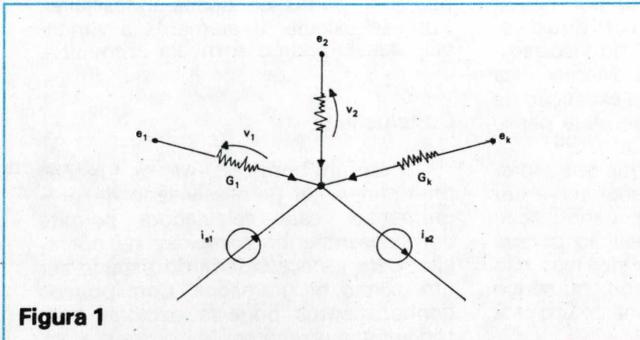



Figura 1

Na aplicação da primeira lei de Kirchoff temos (considerando positivas as correntes cujo sentido de referência sai do nó):

$$-j_1 - j_2 \dots - j_k - j_{51} + i_{52} = 0$$

onde $j_1, j_2, \dots j_k$ são as correntes de ramo facilmente calculadas.

$j_1 = G_1V_1 = G_1(e_1 - e_i)$
$j_2 = G_2V_2 = G_2(e_2 - e_i)$
"
"
"
$j_k = G_kV_k = G_k(e_k - e_i)$

No i $- G_1 e_1 - G_2 e_2 + (G_1 + G_2 + \dots G_k) e_i + \dots - G_k e_k = i_{51} - i_{52}$

Vamos ver um circuito completo (figura 2):

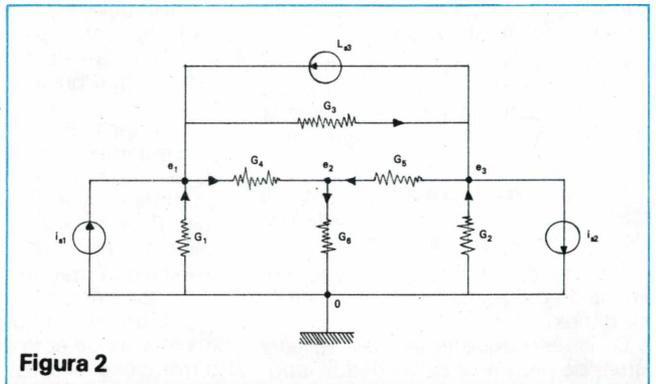



Figura 2

O nó Ø é o nó de referência; e_1, e_2 e e_3 são as tensões nodais dos respectivos nós. As orientações das correntes de ramos foram tomadas em sentidos arbitrários e, como você verá mais adiante, são totalmente dispensáveis.

Aplicando a primeira lei de Kirchoff nos nós 1, 2 e 3, já substituindo as correntes de ramo pela respectiva condutância, vezes a diferença de potencial, temos:

nó 1: $(G_1 + G_3 + G_4) e_1 - G_4 e_2 - G_3 e_3 = i_{51} + i_{53}$

nó 2: $- G_4 e_1 + (G_4 + G_5 + G_6) e_2 - G_5 e_3 = 0$

nó 3: $- G_3 e_1 - G_5 e_2 + (G_2 + G_3 + G_5) e_3 = - i_{52} - i_{53}$

Escrevendo o mesmo sistema de equações em notação matricial, temos:

$$\begin{bmatrix} G_1 + G_3 + G_4 & -G_4 & \\ G_1 + G_3 + G_4 & -G_4 & -G_3 \\ -G_4 & G_4 + G_5 + G_6 & -G_5 \\ -G_3 & -G_5 & G_2 + G_3 + G_5 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} i_{s1} + i_{s3} \\ 0 \\ -i_{s2} - i_{s3} \end{bmatrix}$$

Essa equação é da forma geral

$$G_n \cdot \vec{e} = \vec{i}_{sn}$$

onde G_n = matriz das condutâncias nodais

$\vec{e}$ = vetor das tensões nodais (incógnitas).

$\vec{i}_{sn}$ = vetor das fontes de correntes equivalentes.

Todo o problema consiste então em obter a matriz G_n e o vetor i_{sn}. A regra para se obter diretamente a matriz G_n é muito prática.

No elemento (i,i) é colocada a soma de todas as condutâncias que se ligam ao nó i da rede. No circuito anterior repare que ao nó 1 estão ligadas as condutâncias G_1, G_3, e G_4, logo o elemento (i,i) da matriz tem que ser $G_1 + G_3 + G_4$.

No elemento (i,j) com $i \neq j$, é colocado o negativo da soma das condutâncias que ligam diretamente o nó i ao nó j. No circuito estudado, entre os nós 1 e 2 há a condutância G_4, assim os elementos (1,2) e (2,1), (repare que a matriz é simétrica) são $-G_4$.

Montar o vetor $\vec{i}_{sn}$ é bem fácil. Sua i-ésima linha é dada pela soma das correntes de fontes que chegam ao nó. Sinal positivo é dado à corrente que entra no nó e sinal negativo à corrente que sai.

No circuito anterior, no nó 1 entram as correntes de fonte i_{s1} e i_{s3} e logo a primeira linha é $i_{s1} + i_{s3}$. Já no nó não entra nenhuma corrente de fonte e do nó 3 saem as correntes i_{s2} e i_{s3}. Logo a terceira linha é $-i_{s2} - i_{s3}$.

Veremos agora um exemplo numérico. Vamos mostrar manualmente as matrizes e utilizar o programa MATRIX, do Math Pac, para resolver o sistema de equações associado.

No circuito da figura 3, os valores dados são as condutâncias.

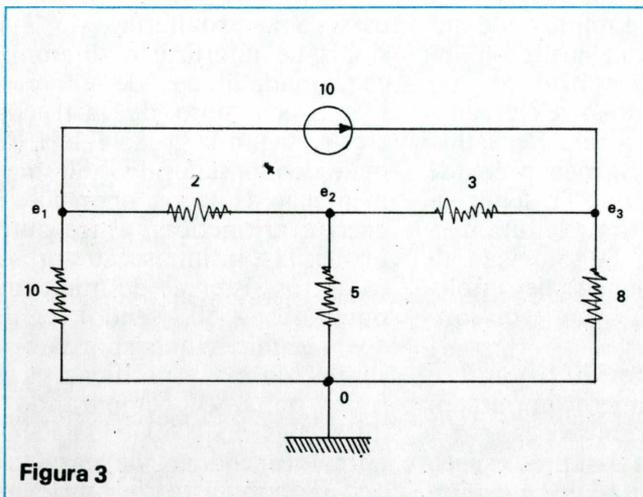



Figura 3

$$G_r = \begin{bmatrix} 10 + 2 & -2 & 0 \\ -2 & 5 + 3 + 2 & -3 \\ 0 & -3 & 3 + 8 \end{bmatrix}$$

$$G_n = \begin{bmatrix} 12 & -2 & 0 \\ -2 & 10 & -3 \\ 0 & -3 & 11 \end{bmatrix}$$

$$i_{sn} = \begin{bmatrix} -10 \\ 0 \\ 10 \end{bmatrix}$$

Assim, o sistema de equações fica:

$$\begin{bmatrix} -12 & -2 & 0 \\ -2 & 10 & -3 \\ 0 & -3 & 11 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} -10 \\ 0 \\ 10 \end{bmatrix}$$

Usemos agora o programa MATRIX

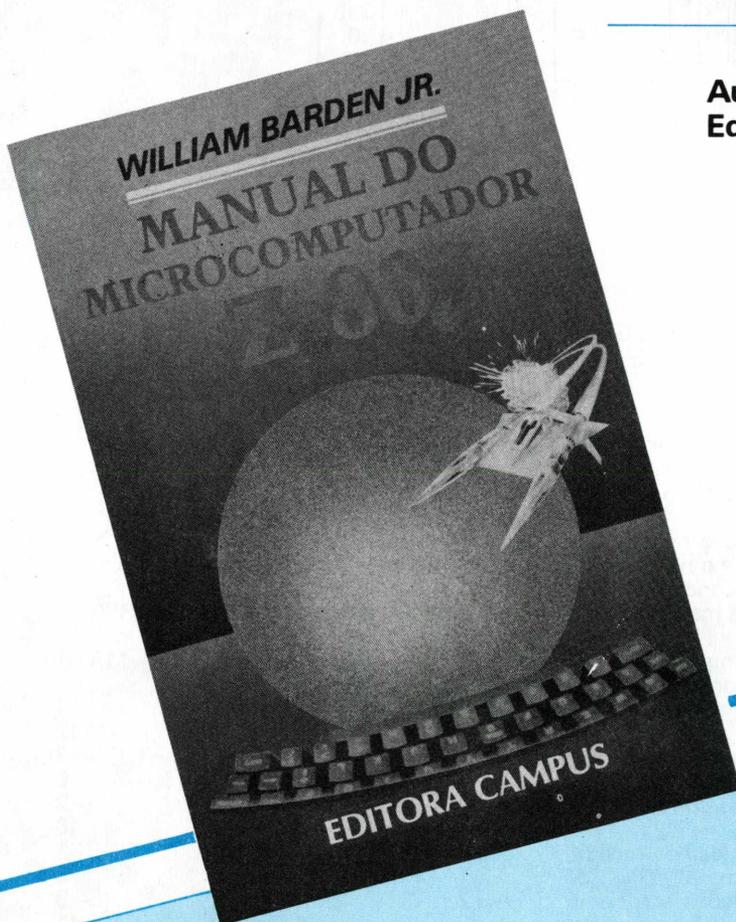
```
XEQ                 MATRIX

ORDER = ?                3        R/S
A 1,1 = ?               12        R/S
A 1,2 = ?              - 2        R/S
A 1,3 = ?                0        R/S
A 2,1 = ?              - 2        R/S
A 2,2 = ?               10        R/S
A 2,3 = ?              - 3        R/S
A 3,1 = ?                0        R/S
A 3,2 = ?              - 3        R/S
A 3,3 = ?               11        R/S

XEQ                  SIMEQ

B1 = ?                - 10        R/S
B2 = ?                   0        R/S
B3 = ?                  10        R/S
R/S«

R/S
X1 = -0,8134          (e1)
R/S
x2 = 0,1199           (e2)
R/S
x3 = 0,9418           (e3)
```

Nos números seguintes discutiremos a análise de circuitos mais complicados e faremos com que, dado o circuito, a calculadora já monte a matriz de condutâncias nodais e o vetor de correntes de fontes.

Bibliografia: Orsini, L. Q. A Análise Nodal e suas variantes, 1984.

LIVROS

Manual do Microcomputador Z80

Autor: William Barden Jr.
Editora Campos

Ana Lúcia Alcântara

A finalidade do livro é fornecer aos leitores toda a informação a respeito do Z-80. Para isto, Barden Jr. dividiu seu texto em seções que ilustram bem as características, arquitetura, entre outros pontos do famoso microprocessador.

Conforme foi ressaltado no prefácio, o autor tem como objetivos básicos "apresentar ao leitor o hardware, examinar os aspectos quase esmagadores (em número de instruções) do software do Z-80 e descrever sistemas de microcomputadores construídos em torno do mesmo"

O autor apresenta desta forma, um potente e requisitado microcomputador, colocando-o inclusive, como substituto do 8080A e referindo-se a ele como "o estado da técnica". Pensando assim, William Barden pretende principalmente, que "o leitor tire grande proveito da leitura deste livro e que o Z-80 solucione alguns dos problemas de implementação de hardware e de software".

Com cerca de 370 páginas, o "Manual do microcomputador Z-80" é dividido em seções — três no total — cujos assuntos específicos são apresentados em capítulos. A primeira parte aborda temas como arquitetura do Z-80, sinais de interface e sincronização, modalidades de endereçamento, conjunto de instruções, etc. A segunda seção fala a respeito do montador do Z-80, movimentação de dados, operações lógicas e aritméticas, entre outros temas. E a última seção apresenta outros sistemas de microcomputadores Z-80, sendo seguida dos apêndices que abordam assuntos sobre as especificações elétricas do Z-80, comparando as instruções do 8080 e do Z-80 e apresenta códigos de caracteres ASCII e uma lista dos fabricantes norte-americanos do mesmo.

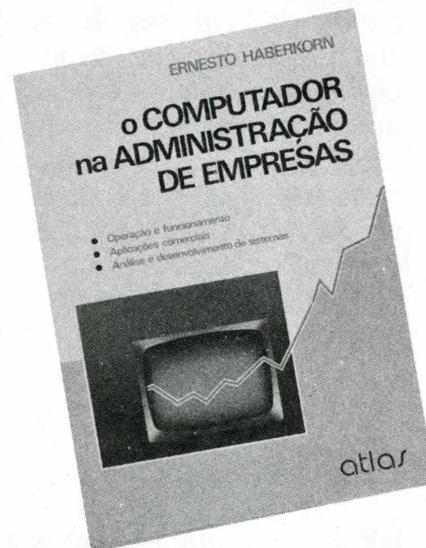
Manutenção de microcomputadores

Autor: Maurício Caruzo Reis
Editora Petit

O livro não pretende, segundo os editores, ser um manual para a elaboração de planejamentos de manutenção de microcomputadores, abordando técnicas preventivas e questões comerciais, mas sim abordar as técnicas de pesquisa de defeitos existentes — através de uma abordagem de âmbito geral — e os instrumentos necessários para suas aplicações.

Em suas 203 páginas, o livro apresenta as técnicas ilustrando as partes pragmáticas através de exemplos explicativos.

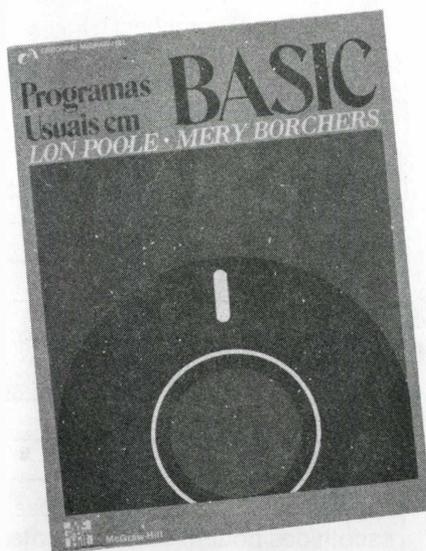
Dividido em 12 capítulos, *Manutenção de Microcomputadores* aborda assuntos como: tópicos de interesse em eletrônica digital, arquitetura e funcionamento de microcomputadores, microprocessador Z-80, auto-teste, entre outros. **A.L.A.**

O Computador na Administração de Empresas

Autor: Ernesto Haberkorn
Editora: Atlas

Quem espera encontrar neste livro programas complicados de computação, pode desistir. Ernesto Haberkorn está preocupado em introduzir o leigo no mundo da informática e para isso criou um livro que poderia ser considerado o be-a-bá em computação.

Trata-se dos primeiros passos que um leigo deve dar para se ver envolvido no mundo da computação. De forma clara e simples, o autor aborda os termos que envolvem a informática de modo geral, apresenta as características do computador, sua forma de utilização na empresa, escritório e cotidiano do profissional. Explica, de forma comparativa, como funcionam os software, para que servem e como usá-los, diferencia automação e mecanização e finaliza o livro com um estudo de sistema integrado, para desmistificar o que tem sido a razão de muitos fracassos nas empresas: a automação de todas atividades administrativa de forma integrada.

Autor de outros dois livros — Introdução à Análise de Sistemas e Computador e Processamento de Dados — Haberkorn direciona esta obra aos principiantes e principalmente aos empresários que já se conscientizaram de que a automação deixou de ser futuro. **S.A.M.**

Programas Usuais em BASIC

Autores: Lon Poole e Mery Borchers
Editora MacGrawHill

Constituído de 74 programas com aplicações para a área financeira, *Programas usuais em BASIC,* tem o objetivo de fornecer aos usuários de microcomputadores, programas escritos dentro de uma série restrita de instruções de maneira a torná-los compatíveis com as diversas versões de BASIC. Para isto, os autores preocuparam-se em descrever cada programa, incluindo exemplos de aplicação junto com a listagem dos mesmos e ilustrando os dados obtidos durante a sua execução nos exemplos apresentados.

Para possibilitar maior campo de aplicação, os autores incluiram outras opções que sugerem formas de alteração através de descrição, exemplos e listagens parciais para cada opção extra. **A.L.A.**

Lig-4 & Jogo da Memória

Maurício de Albuquerque e Silva

Lig-4

O programa **Lig-4** é uma nova adaptação para o TK do jogo também conhecido como Seqüência 4, 4 Numa Fila, etc. É jogado por 2 pessoas e o objetivo é conseguir uma seqüência, em qualquer direção, de 4 peças. Na sua vez, o jogador deverá digitar a letra correspondente à coluna onde ele quer colocar sua peça. A peça será colocada na primeira disponível de baixo para cima. O próprio TK-82, à cada rodada, verifica se há um vencedor.

Possíveis alterações:

Linha 145

Código do caractere do jogador 1

Let X = 128-120 * (Q = 1)

Código do caractere do jogador 2

Linha 25Ø

FAST — totalmente opcional; apenas dá mais dinamismo ao jogo.

LIG-4
Memória ocupada = 2319 bytes
Soma Sintática = 9211
Nível = 1

MEMÓRIA
Memória ocupada = 3157 bytes
Soma Sintática = 46912
Nível = 1

Jogo da Memória

Este outro programa, **Jogo da Memória**, é uma versão do conhecido jogo, cujo objetivo é achar o maior número de pares possível. Ao ser rodado, o programa entra por pouco tempo em modo FAST para criação do painel e embaralhamento das peças. O computador então pedirá o número de jogadores. Na sua vez, o jogador deverá entrar com a posição das 2 peças que ele quer que sejam descobertas. A posição da peça é dada por uma string formada, na ordem, pelo número da linha e pela letra da coluna. Se formar um par, ganhará um ponto e jogará outra vez; caso contrário, passará a vez ao próximo. Para terminar a partida antes dos 49 pares serem achados basta interromper o programa e digitar GOTO 63Ø.

Possíveis alterações:

Linha 5Ø

Contém os 49 pares usados no jogo. Você não deve usar pares com os caracteres CHR$ Ø e CHR$ 128.

Linhas 16Ø e 18Ø

Contêm o número máximo de jogadores.

Linha 335

Indica o tempo que as peças escolhidas podem ser vistas antes de serem cobertas.

```
   0 REM PROGRAMA LIG-4
   0 REM MAURICIO ALBUQUERQUE E
SILVA
  10 LET W=PEEK 16396+256*PEEK 1
6397+1
  20 CLS
  30 PRINT
  40 LET A$=" +-+-+-+-+-+-+-+-
+-+-+-+-+-+  "
  50 LET B$=" I I I I I I I I
I I I I I  "
  60 FOR N=1 TO 9
  70 PRINT A$;B$
  80 NEXT N
  90 PRINT A$
 100 FOR N=1 TO 14
 110 PRINT TAB 2*N;CHR$ (37+N);
 120 NEXT N
 125 LET J=0
 130 LET Q=1
 140 LET P=W+594
 145 LET X=128-120*(Q=1)
 150 PRINT AT 0,2;"JOGADOR ";Q;"
: SUA VEZ DE JOGAR"
 160 LET S$=INKEY$
 170 IF S$<"A" OR S$>"N" THEN GO
TO 160
 180 LET P=P+2*(CODE S$-37)
 190 IF NOT PEEK P THEN GOTO 230
 200 LET P=P-66
 210 GOTO 190
 230 IF P<W THEN GOTO 140
 240 POKE P,X
 245 LET J=J+1
 250 FAST
 270 LET X$="32333401"
 280 FOR M=1 TO 4
 285 LET S=0
 290 LET Z=VAL X$((2*M-1) TO (2*
M))
 300 GOSUB 1000
 310 LET Z=-Z
 320 GOSUB 1000
 330 IF S=3 THEN GOTO 490
 340 NEXT M
 350 SLOW
 360 IF J=126 THEN GOTO 400
 370 LET Q=1*(Q=2)+2*(Q=1)
 380 GOTO 140
 400 PRINT AT 0,0;"           PARTI
DA EMPATADA           "
 410 PRINT AT 21,1;"TECLE <N/L>
PARA OUTRA PARTIDA"
 420 IF INKEY$<>CHR$ 118 THEN GO
TO 420
 430 RUN
 490 SLOW
 500 PRINT AT 0,0;"    O JOGADOR
";Q;" E O VENCEDOR      "
 510 GOTO 410
1000 LET T=P+2*Z
1010 GOSUB 2000
1020 RETURN
2000 FOR N=1 TO 3
2010 IF PEEK T<>X THEN RETURN
2020 LET S=S+1
2030 LET T=T+2*Z
2040 NEXT N
2050 RETURN
2060 SAVE "LIG-4"
2070 RUN
```

```
   0 REM PROGRAMA JOGO DA MEMORI
A
   0 REM MAURICIO ALBUQUERQUE E
SILVA
  10 CLS
  15 LET W=0
  20 LET P=PEEK 16396+256*PEEK 1
6397+1
  25 DIM Y(2)
  30 DIM X(2)
  35 FAST
  40 RAND
  50 LET X$=" AABBCCDDEEFFGGHHII
JJKKLLMMNNOOPPQQRRSSTTUUVVWWXXYY
ZZ00112233445566778899££$$??<<>>
==--++**//;;,,((  "
  60 DIM Y$(7,14)
  70 FOR N=1 TO 7
  75 PRINT AT 2*N,0;CHR$ (28+N)
  80 FOR M=1 TO 14
  90 LET B=LEN X$-2
 100 LET S=INT (RND*B+2)
 110 LET Y$(N,M)=X$(S)
 120 LET X$=X$( TO (S-1))+X$((S+
1) TO )
 130 PRINT AT 2*N,2*M;"■"
 135 IF N=7 THEN PRINT AT 0,2*M;
CHR$ (37+M)
 140 NEXT M
 150 NEXT N
 160 PRINT AT 16,1;"QUANTOS VAO
JOGAR ?(MAXIMO=6)"
 165 SLOW
 170 LET A$=INKEY$
 180 IF A$<"2" OR A$>"6" THEN GO
TO 170
 190 LET N=VAL A$
 195 DIM S(N)
 200 FOR G=1 TO N
 210 LET S(G)=0
 220 NEXT G
 230 LET Q=1
 240 PRINT AT 16,0;"  JOGADOR ";
Q;": SUA VEZ DE JOGAR   "
 245 DIM C$(2)
 250 FOR D=1 TO 2
 255 INPUT B$
 260 IF LEN B$<>2 THEN GOTO 255
 265 IF B$(1)>"9" THEN GOTO 255
 270 LET X(D)=CODE B$(1)-28
 275 IF B$(2)<"A" THEN GOTO 255
 280 LET Y(D)=CODE B$(2)-37
 285 IF PEEK (P+66*X(D)+2*Y(D))<
>128 THEN GOTO 255
 290 PRINT AT 2*X(D),2*Y(D);Y$(X
(D),Y(D))
 300 LET C$(D)=Y$(X(D),Y(D))
 305 NEXT D
 310 IF C$(1)=C$(2) THEN GOTO 50
0
 315 PRINT AT 18,10;"  ERROU   "
 320 LET J=1
 325 LET Q=Q+1
 330 IF Q>N THEN LET Q=1
 335 FOR H=1 TO 75
 340 NEXT H
 345 PRINT AT 2*X(1),2*Y(1);CHR$
(128*J)
 350 PRINT AT 2*X(2),2*Y(2);CHR$
(128*J)
 375 PRINT AT 18,10;"           "
 380 GOTO 240
 500 PRINT AT 18,10;" ACERTOU "
 510 LET S(Q)=S(Q)+1
 515 LET J=0
 520 LET W=W+1
 530 IF W=49 THEN GOTO 600
 540 GOTO 335
 600 PRINT AT 20,7;"FINAL DE PAR
TIDA"
 610 FOR H=1 TO 75
 620 NEXT H
 630 CLS
 640 PRINT AT 2,7;"RESULTADO FIN
AL"
 650 FOR H=1 TO N
 660 PRINT AT 3+H,0;"JOGADOR ";H
;": ";S(H);" PAR";"ES" AND (S(H)
>1)
 670 NEXT H
 680 PRINT AT 5+N,1;"TECLE <N/L>
PARA OUTRA PARTIDA"
 690 IF INKEY$<>CHR$ 118 THEN GO
TO 690
 700 RUN
 710 SAVE "JM"
 720 RUN
```

Resposta ao Quebra-Cabeça

A Torre de Vogel no TK 2000

O quebra-cabeça "A Torre de Vogel", proposto na revista 15, teve uma resposta publicada na Microhobby nº 18. Entretanto, logo após termos escolhido a melhor resposta, surgiu na redação uma carta contendo a solução para o TK 2000. Em virtude de nunca termos recebido soluções para Quebra-Cabeças para o TK 2000, resolvemos publicar também esta solução, enviada por Alberto Alves da Motta, de São Paulo.

A solução

"Procurei resolver o quebra-cabeça 'A Torre de Vogel' e montei dois programas, um deles bastante simples, com apenas 8 linhas, que, porém, levou aproximadamente 14,33 segundos para fornecer os resultados com a precisão que desejei.

Descontente com a demora, fiz outro programa com 28 linhas e o aparelho forneceu os mesmos resultados, com a precisão desejada, porém mais rápido e terminou os cálculos em 3,44 segundos.

O programa baseou-se na seqüência de cálculos da equação que fornece o comprimento da escada em relação ao ângulo que ela faz om o chão (N):

$$x = \frac{2}{\text{sen } N} + \frac{8}{\cos N}$$

Sendo x o comprimento da escada e N o ângulo da escada com o chão."

Programa I

```
5  REM  TORRE DE VOGEL
10 X = 1000000
20  FOR I = 1 TO 89 STEP
0.01
30 R = 180 / 3.141592
40 X(1) = 2 /  SIN (I / R
) + 8 /  COS
(I / R)
50  IF X(1) > X GOTO 80
60 X = X(1): NEXT
70  PRINT "ALTURA DA TORR
E (ESCADA)=";
X: PRINT
80  PRINT "ANGULO DE ENTR
ADA DA ESCADA
=";I - 0.01
```

Programa II

```
5  REM  TORRE DE VOGEL
10 X = 1000000
20  FOR I = 0.01 TO 89.99
 STEP 10
30 R = 180 / 3.141592
40 X(1) = 2 /  SIN (I / R
) + 8 /  COS
(I / R)
50  IF X(1) > X GOTO 190
60 X = X(1): NEXT
70  PRINT "ALTURA=";X: PR
INT
80  PRINT "ANGULO ENTRADA
=";M - 0.01: END
190 X = X(1)
200  FOR J = I TO 0.01 ST
EP  - 2
210 X(1) = 2 /  SIN (J /
R) + 8 /  COS
(J / R)
220  IF X(1) > X GOTO 300

230 X = X(1): NEXT
300 X = X(1)
310  FOR K = J TO 89.99 S
TEP 0.5
320 X(1) = 2 /  SIN (K /
R) + 8 /  COS
(K / R)
330  IF X(1) > X GOTO 400

340 X = X(1): NEXT
400 X = X(1)
410  FOR L = K TO 0.01 ST
EP  - 0.1
420 X(1) = 2 /  SIN (L /
R) + 8 /  COS
(L / R)
430  IF X(1) > X GOTO 500

440 X = X(1): NEXT
500 X = X(1)
510  FOR M = L TO 89.99 S
TEP 0.01
520 X(1) = 2 /  SIN (M /
R) + 8 /  COS
(M / R)
530  IF X(1) > X GOTO 80
540 X = X(1): NEXT
```

Resultados

Em ambos os programas, obtive:
altura: x = 13,2073221 ≅ 13,20 m
ângulo de entrada: N = 32,21°

Chegou a mais alta patente em videogame.

Onyx Junior. Uma verdadeira batalha de emoções.

Prepare-se. Dentro de sua própria casa, você vai perseguir e abater mísseis e tanques de guerra. Seus jatos escaparão por milímetros dos potentes canhões inimigos, planetas explodirão em chamas. Você vai conhecer, nas cores mais dramáticas, na maior nitidez, as sensações de uma verdadeira batalha.

Entregue-se. Você simplesmente não vai resistir a fantástica aventura que é ter um Onyx Junior.

O mais moderno, o mais completo videogame que você já viu (o único neste sistema que tem pause), com mais de 300 jogos diferentes: o Onyx Junior usa todos os cartuchos da Linha Atari®.

Apresente-se. A mais alta patente em videogame espera você para um encontro inesquecível.

MICRODIGITAL

Apresentamos o TK 2000 II. Ele roda o programa mais famoso do mundo.

De hoje em diante nenhuma empresa, por menor que seja, pode dispensar o TK 2000 II. Por que?

O novo TK 2000 II roda o Multicalc: a versão Microsoft do Visicalc®, o programa mais famoso em todo o mundo.

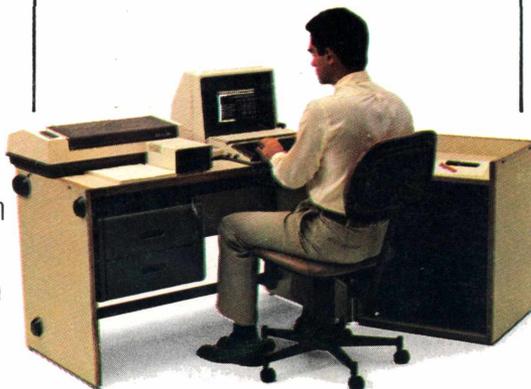
Isto significa que, com ele, você controla estoques, custos, contas a pagar, faz sua programação financeira, efetua a folha de pagamentos e administra minuto a minuto as suas atividades.

Detalhe importante: o novo TK 2000 II, com Multicalc, pode intercambiar planilhas com computadores da linha Apple®.

E, como todo business computer que se preza, ele tem teclado profissional, aceita monitor, diskette, impressora e já vem com interface.

Além de poder ser ligado ao seu televisor (cores ou P&B), oferecendo som e imagem da melhor qualidade.

Portanto, peça logo uma demonstração do novo TK 2000 II, nas versões 64K ou 128K de memória.

A mais nova estrela do show business só espera por isto para estrear no seu negócio.

Preço de lançamento* (128 K):
Cr$ 1.949.850

MICRODIGITAL
computadores pessoais

Open for Business.

® Marca registrada da Apple Computer.

Filiada à ABICOMP

® Marca registrada da Visicorp.

* Sujeito a alteração sem prévio aviso.

CERTIFICADO ESPECIAL DE RESERVA

VÁLIDO ATÉ 30.06.85

AUTORIZO PELO PRESENTE MINHA ☐ ASSINATURA INICIAL ☐ RENOVAÇÃO ☐ ALTERAÇÃO DE ENDEREÇO DA REVISTA MICROHOBBY (12 EDIÇÕES).

ANEXO REMETO ☐ VALE POSTAL Nº ________ ☐ CHEQUE Nº ________ BCO. ________

NOMINAL À MICROMEGA P.M.D. LTDA. NO VALOR DE CR$ ________

NOME ________

ENDEREÇO ________

CIDADE ________ BAIRRO ________ CEP ________

FONE ________ ESTADO ________

PREÇOS: ASSINATURA: Cr$ 60.000
RENOVAÇÃO: Cr$ 50.000

Em caso de renovação de assinatura colar neste campo a etiqueta de endereçamento atual.

Micromega P.M.D. Ltda. • Av. Angélica, 2318 • 14 And. • São Paulo • Cep 01296 • Caixa Postal 54096 • Fone: 255-0366